

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS  
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



***Trabajo Fin de Grado***

**Caracterización experimental del  
rendimiento del estándar 802.11n en un  
entorno doméstico**  
(Experimental characterization of 802.11n  
performance in home networks)

Para acceder al Título de

***Graduado en  
Ingeniería de Tecnologías de Telecomunicación***

Autor: Rodrigo Gañán de Andrés

Abril - 2019



E.T.S. DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACION

## **GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

### **CALIFICACIÓN DEL TRABAJO FIN DE GRADO**

**Realizado por:** Rodrigo Gañán de Andrés

**Director del TFG:** Marta García Arranz

**Título:** “Caracterización experimental del rendimiento del estándar 802.11n  
en un entorno doméstico”

**Title:** “Experimental characterization of 802.11n performance in home  
networks “

**Presentado a examen el día:** 30/04/2019

para acceder al Título de

## **GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

### Composición del Tribunal:

Presidente (Apellidos, Nombre): Sanz Gil, Roberto

Secretario (Apellidos, Nombre): Alcalá Galán, Francisco José

Vocal (Apellidos, Nombre): García Arranz, Marta

Este Tribunal ha resuelto otorgar la calificación de: .....

Fdo.: El Presidente

Fdo.: El Secretario

Fdo.: El Vocal

Fdo.: El Director del TFG

Vº Bº del Subdirector

Trabajo Fin de Grado Nº

## **Resumen**

---

El Grupo de Ingeniería Telemática (GIT) ha venido realizando numerosos trabajos en el campo de la caracterización del rendimiento de los estándares IEEE 802.11 para redes de área local inalámbrica (WLAN). Hasta la fecha se disponía de un conocimiento profundo de cómo era dicho comportamiento para las primeras especificaciones a/b/g del estándar; sin embargo, la norma 802.11n, aprobada en 2009 y de uso mayoritario actualmente, introduce importantes mejoras, tanto en la capa física como en la capa MAC, por lo que se ha hecho necesario ampliar dicha caracterización. Además, se ha decidido abordar este trabajo en un entorno real como es el de una red Wi-Fi en un entorno doméstico.

## **Abstract**

---

The Engineering Telematics Group (GIT) have been carried out numerous works in the field of performance characterization about the standards IEEE 802.11 for the wireless local area networks (WLAN). So far, a deep knowledge had been provided about how the performance for the first specifications a/b/g of the standard was; nevertheless, 802.11n specification, approved in 2009 and of broad use nowadays, introduce very important improvements on the physical layer as well as on the MAC layer. Therefore, it has been necessary to expand this characterization. In addition, this work has been done in a real environment as it could be a Wi-Fi network in a home network.

# Índice General

---

Capítulo 1. Introducción.....	1
1.1 Presentación y objetivos.....	1
1.2 Estructura de la memoria .....	3
Capítulo 2. Conocimientos previos .....	4
2.1 Estándar 802.11-1997 .....	4
2.2 Estándar 802.11b .....	4
2.3 Estándar 802.11a.....	4
2.4 Estándar 802.11g.....	5
2.5 Estándar 802.11n .....	5
2.6 Agregación de tramas en el estándar 802.11n .....	9
2.6.1 A-MSDU.....	9
2.6.2 A-MPDU .....	9
Capítulo 3. Descripción de la plataforma de caracterización .....	11
3.1 Elementos .....	11
3.1.1 Transmisor .....	11
3.1.2 Router .....	11
3.1.3 Receptor.....	12
3.1.4 Adaptador USB Wi-Fi .....	12
3.2 Escenario.....	13
3.3 Tráfico interferente en el escenario .....	15
Capítulo 4. Configuración de los equipos.....	17
4.1. Crear partición e instalación Ubuntu .....	17
4.2 Herramientas de medida .....	18
4.2.1 Nttcp .....	18
4.2.2 Iperf .....	19
4.3 Programas para análisis, obtención y representación de datos .....	19
4.4 Instalación programas .....	20
4.4.1 Instalación programas en Ubuntu .....	20
4.4.2 Instalación programas en Windows .....	21
4.5 Características internas adaptador USB Wi-Fi TL-WN821N.....	21
Capítulo 5. Realización de las medidas .....	22
5.1 Creación red Wi-Fi en modo Ad-hoc .....	22
5.2 Captura datos.....	24
5.2.1 Modo monitor .....	24

5.2.2 Wireshark.....	25
5.2.3 Medidas con Nttcp .....	26
5.2.4 Filtrado.....	27
5.3 Obtención datos con C .....	30
5.3.1 Lectura y extracción de datos .....	30
5.3.2 Escritura de resultados .....	38
5.4 Matlab .....	41
5.5 Medidas con lperf .....	48
Capítulo 6. Análisis de los resultados.....	51
6.1 Medidas en modo infraestructura .....	51
6.1.1 Análisis medidas escenario 1 .....	53
6.1.2 Análisis medidas escenario 2 .....	63
6.1.3 Análisis medidas escenario 3 .....	71
6.1.4 Análisis medidas escenario 4 .....	78
6.1.5 Resultados lperf en modo infraestructura.....	85
6.2 Medidas en modo Ad-hoc.....	86
6.2.1 Análisis medidas escenario 1 .....	86
6.2.2 Análisis medidas escenario 2 .....	90
6.2.3 Análisis medidas escenario 3 .....	93
6.2.4 Análisis medidas escenario 4 .....	96
6.2.5 Resultados lperf en modo Ad-hoc .....	100
6.2.6 Entorno ideal .....	101
Capítulo 7. Conclusiones y líneas de trabajo futuro .....	105
Bibliografía .....	106

## Índice de Figuras

---

Figura 1.1. Red Wi-Fi en modo Infraestructura .....	2
Figura 1.2. Red Wi-Fi en modo Ad-hoc .....	2
Figura 2.1. Channel bonding.....	6
Figura 2.2. MIMO.....	6
Figura 2.3. Características obligatorias frente a las características opcionales del estándar 802.11n .....	8
Figura 2.4. Funcionamiento agregación A-MSDU .....	9
Figura 2.5. Funcionamiento agregación A-MPDU.....	10
Figura 3.1. Router Arcadyan prv3399b-b-It.....	12
Figura 3.2. Adaptador inalámbrico TP-LINK modelo TL-WN821N .....	13
Figura 3.3. Plano aproximado del entorno.....	13
Figura 3.4. Escenario con red Wi-Fi en modo Ad-hoc .....	14
Figura 3.5. Interfaz programa inSSIDer Home .....	15
Figura 4.1. Partición de 50 GB del ordenador portátil .....	17
Figura 4.2. Interfaz del programa Rufus.....	18
Figura 4.3. Iconos de Wireshark, Dev-C++ y Matlab en este orden.....	19
Figura 4.4. Respuesta al comando lsusb .....	21
Figura 4.5. Respuesta al comando “dmesg   grep ‘ath’” .....	21
Figura 5.1. Cuadro de opciones para la creación de una red Wi-Fi .....	23
Figura 5.2. Lista de redes Wi-Fi registradas en el ordenador .....	23
Figura 5.3. Comprobación de la creación de la red Wi-Fi Ad-hoc .....	24
Figura 5.4. SSID de la red Wi-Fi y algunas características .....	25
Figura 5.5. Interfaz del Wireshark con la Wireless Toolbar.....	25
Figura 5.6. Resultados medida nntcp.....	27
Figura 5.7. Filtros empleados .....	28
Figura 5.8. Formato de la trama en Wireshark.....	29
Figura 5.9. Vista del usuario al utilizar el programa .....	30
Figura 5.10. Ciclo While para la lectura del archivo.....	30
Figura 5.11. Primeras dos líneas de cada trama .....	31
Figura 5.12. Bloque de código para conocer tamaño de la trama.....	31
Figura 5.13. Bloque de código para conocer la velocidad de transmisión de la A-MPDU .....	32
Figura 5.14. Bloque de código para conocer la potencia de señal recibida de la A-MPDU .....	33
Figura 5.15. Bloque de código para saber si la trama analizada es una retransmisión .....	33
Figura 5.16. Cabecera IEEE 802.11 de una retransmisión.....	34

Figura 5.17. Bloque de código para crear una lista de todas las tramas ordenadas por orden aparición e indicando el número de veces que se retransmite.....	35
Figura 5.18. Bloque de código para asignar la velocidad de transmisión y potencia de señal a cada trama en la lista de tramas mencionada anteriormente .....	36
Figura 5.19. Bloque de código para indicar en la lista de tramas si una trama es una “retransmisión fantasma” .....	37
Figura 5.20. Bloque de código para indicar el comienzo de la siguiente trama.....	37
Figura 5.21. Bloque de código para escribir el archivo “SolSen.txt” .....	38
Figura 5.22. Visualización del archivo SolSen.txt.....	38
Figura 5.23. Bloque de código para escribir el archivo “SolRate.txt” .....	39
Figura 5.24. Visualización del archivo SolRate.txt .....	39
Figura 5.25. Bloque de código para escribir el archivo “SolRtxflujo.txt” .....	39
Figura 5.26. Visualización del archivo SolRtxflujo.txt .....	40
Figura 5.27. Bloque de código para escribir el archivo “Relaciones.txt” .....	40
Figura 5.28. Visualización del archivo Relación.txt.....	41
Figura 5.29. Lectura archivo SolSen.txt .....	42
Figura 5.30. Lectura archivo SolRate.txt .....	42
Figura 5.31. Lectura archivo Relaciones.txt.....	42
Figura 5.32. Bloque de código para la representación de la potencia de señal de cada A-MPDU .....	42
Figura 5.33. Representación de la potencia de señal de cada A-MPDU .....	42
Figura 5.34. Bloque de código para la representación de la velocidad de transmisión de cada A-MPDU.....	43
Figura 5.35. Representación de la velocidad de transmisión de cada A-MPDU.....	43
Figura 5.36. Bloque de código para la obtención del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad .....	44
Figura 5.37. Bloque de código para la representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad.....	44
Figura 5.38. Representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad.....	45
Figura 5.39. Representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una única velocidad .....	45
Figura 5.40. Bloque de código para la obtención de la probabilidad de transmisión según la velocidad de transmisión .....	46
Figura 5.41. Bloque de código para la representación de la probabilidad de transmisión según la velocidad de transmisión.....	46
Figura 5.42. Representación de la probabilidad de transmisión según la velocidad de transmisión.....	46

Figura 5.43. Bloque de código para la obtención del número de veces que se retransmite cada trama.....	47
Figura 5.44. Bloque de código para la obtención del número de veces que se transmite una retransmisión.....	48
Figura 5.45. Bloque de código para la representación del número de veces que se transmite una retransmisión.....	48
Figura 5.46. Representación del número de veces que se transmite una retransmisión.....	48
Figura 5.47. Resultados de Iperf en el ordenador receptor .....	50
Figura 5.48. Resultados de Iperf en el ordenador transmisor.....	50
Figura 6.1. Formato trama Beacon en Wireshark.....	51
Figura 6.2. Campo información sobre regulación del país.....	52
Figura 6.3. Características opcionales de 802.11n que ofrece mi router .....	52
Figura 6.4. Formato trama “probe request” del dispositivo TP-Link .....	52
Figura 6.5. Características opcionales de 802.11n que ofrece el adaptador Wi-Fi USB TL-WN821N .....	53
Figura 6.6. Potencia de señal recibida en el transmisor en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura .....	57
Figura 6.7. Velocidad de transmisión en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura .....	58
Figura 6.8. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura .....	58
Figura 6.9. Probabilidad de retransmisión a una determinada velocidad en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura .....	59
Figura 6.10. Número de veces que se transmite una retransmisión en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura .....	60
Figura 6.11. Potencia de señal recibida en el transmisor en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	60
Figura 6.12. Velocidad de transmisión en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	61
Figura 6.13. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	61
Figura 6.14. Probabilidad de retransmisión a una determinada velocidad en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura.....	62
Figura 6.15. Número de veces que se transmite una retransmisión en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	62
Figura 6.16. Potencia de señal recibida en el transmisor en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura .....	66
Figura 6.17. Velocidad de transmisión en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura .....	66



Figura 6.18. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura .....	67
Figura 6.19. Probabilidad de retransmisión a una determinada velocidad en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura .....	67
Figura 6.20. Número de veces que se transmite una retransmisión en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura .....	68
Figura 6.21. Potencia de señal recibida en el transmisor en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	68
Figura 6.22. Velocidad de transmisión en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	69
Figura 6.23. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	69
Figura 6.24. Probabilidad de retransmisión a una determinada velocidad en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	70
Figura 6.25. Número de veces que se transmite una retransmisión en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	70
Figura 6.26. Potencia de señal recibida en el transmisor en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura .....	73
Figura 6.27. Velocidad de transmisión en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura .....	73
Figura 6.28. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura .....	74
Figura 6.29. Probabilidad de retransmisión a una determinada velocidad en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura .....	74
Figura 6.30. Número de veces que se transmite una retransmisión en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura .....	75
Figura 6.31. Potencia de señal recibida en el transmisor en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	75
Figura 6.32. Velocidad de transmisión en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	76
Figura 6.33. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	76
Figura 6.34. Probabilidad de retransmisión a una determinada velocidad en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	77
Figura 6.35. Número de veces que se transmite una retransmisión en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	77
Figura 6.36. Potencia de señal recibida en el transmisor en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura .....	80
Figura 6.37. Velocidad de transmisión en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura .....	80

Figura 6.38. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura .....	81
Figura 6.39. Probabilidad de retransmisión a una determinada velocidad en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura .....	81
Figura 6.40. Número de veces que se transmite una retransmisión en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura .....	82
Figura 6.41. Potencia de señal recibida en el transmisor en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	82
Figura 6.42. Velocidad de transmisión en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	83
Figura 6.43. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	83
Figura 6.44. Probabilidad de retransmisión a una determinada velocidad en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	84
Figura 6.45. Número de veces que se transmite una retransmisión en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura .....	84
Figura 6.46. Potencia de señal recibida en el transmisor en el escenario 1 en una red Wi-Fi en modo Ad-hoc .....	87
Figura 6.47. Velocidad de transmisión en el escenario 1 en una red Wi-Fi en modo Ad-hoc .....	88
Figura 6.48. Probabilidad de retransmisión a una determinada velocidad en el escenario 1 en una red Wi-Fi en modo Ad-hoc .....	88
Figura 6.49. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1 en una red Wi-Fi en modo Ad-hoc .....	89
Figura 6.50. Número de veces que se transmite una retransmisión en el escenario 1 en una red Wi-Fi en modo Ad-hoc .....	89
Figura 6.51. Potencia de señal recibida en el transmisor en el escenario 2 en una red Wi-Fi en modo Ad-hoc .....	90
Figura 6.52. Potencia de señal recibida en el receptor en el escenario 2 en una red Wi-Fi en modo Ad-hoc .....	91
Figura 6.53. Velocidad de transmisión en el escenario 2 en una red Wi-Fi en modo Ad-hoc .....	91
Figura 6.54. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2 en una red Wi-Fi en modo Ad-hoc .....	92
Figura 6.56. Número de veces que se transmite una retransmisión en el escenario 2 en una red Wi-Fi en modo Ad-hoc .....	93
Figura 6.57. Potencia de señal recibida en el transmisor en el escenario 3 en una red Wi-Fi en modo Ad-hoc .....	94
Figura 6.58. Velocidad de transmisión en el transmisor en el escenario 3 en una red Wi-Fi en modo Ad-hoc .....	94
Figura 6.59. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3 en una red Wi-Fi en modo Ad-hoc .....	95

Figura 6.60. Probabilidad de retransmisión a una determinada velocidad en el escenario 3 en una red Wi-Fi en modo Ad-hoc.....	95
Figura 6.61. Número de veces que se transmite una retransmisión en el escenario 3 en una red Wi-Fi en modo Ad-hoc .....	96
Figura 6.62. Potencia de señal recibida en el transmisor en el escenario 4 en una red Wi-Fi en modo Ad-hoc .....	97
Figura 6.63. Potencia de señal recibida en el receptor en el escenario 4 en una red Wi-Fi en modo Ad-hoc .....	97
Figura 6.64. Velocidad de transmisión en el transmisor en el escenario 4 en una red Wi-Fi en modo Ad-hoc .....	98
Figura 6.65. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4 en una red Wi-Fi en modo Ad-hoc .....	98
Figura 6.66. Probabilidad de retransmisión a una determinada velocidad en el escenario 4 en una red Wi-Fi en modo Ad-hoc.....	99
Figura 6.67. Número de veces que se transmite una retransmisión en el escenario 4 en una red Wi-Fi en modo Ad-hoc .....	99
Figura 6.68. Potencia de señal recibida en el transmisor en una red Wi-Fi en modo Ad-hoc en un entorno ideal .....	101
Figura 6.69. Velocidad de transmisión en el transmisor en una red Wi-Fi en modo Ad-hoc en un entorno ideal .....	102
Figura 6.70. Probabilidad de retransmisión a una determinada velocidad en una red Wi-Fi en modo Ad-hoc en un entorno ideal.....	102
Figura 6.71. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en una red Wi-Fi en modo Ad-hoc en un entorno ideal.....	103
Figura 6.72. Número de veces que se transmite una retransmisión en una red Wi-Fi en modo Ad-hoc en un entorno ideal.....	103

## Índice de tablas

---

Tabla 2.1. Lista velocidades de transmisión del estándar 802.11n .....	7
Tabla 6.1. Escenario 1: Transmisor y Router juntos por la mañana en una red Wi-Fi en modo infraestructura.....	56
Tabla 6.2. Escenario 2: Transmisor y Receptor juntos por la mañana alejados del router en una red Wi-Fi en modo infraestructura .....	64
Tabla 6.3. Escenario 3: Transmisor y router juntos por la tarde alejados del receptor en una red Wi-Fi en modo infraestructura .....	72
Tabla 6.4. Escenario 4: Transmisor y Receptor juntos por la tarde alejados del router en una red Wi-Fi en modo infraestructura .....	79
Tabla 6.5. Medidas Iperf realizadas en el escenario 1 con un tiempo de 3 segundos.....	85
Tabla 6.6. Medidas Iperf realizadas en el escenario 4 con un tiempo de 8 segundos.....	85
Tabla 6.7. Resultados de las medidas en el escenario 1 en una red Ad-hoc.....	87
Tabla 6.8. Resultados de las medidas en el escenario 2 en una red Ad-hoc.....	90
Tabla 6.9. Resultados de las medidas en el escenario 3 en una red Ad-hoc.....	93
Tabla 6.10. Resultados de las medidas en el escenario 4 en una red Ad-hoc.....	96
Tabla 6.11. Medidas Iperf realizadas en el escenario 1 con un tiempo de 3 segundos.....	100
Tabla 6.12. Medidas Iperf realizadas en el escenario 4 con un tiempo de 8 segundos.....	100

## Acrónimos

---

**IEEE:** Institute of Electrical and Electronics Engineers

**IP:** Internet protocol

**RSSI:** Received Signal Strength Indicator

**SOC:** System On a Chip

**USB:** Universal Serial Bus

**UDP:** User Datagram Protocol

**Wi-Fi:** Wireless Fidelity

**MPDU:** MAC Protocol Data Unit

**MSDU:** MAC Service Data Unit

**A-MPDU:** Aggregated MPDU

**A-MSDU:** Aggregated MSDU

**MIMO:** Multiple-input Multiple-output

**PER:** Packet Error Rate

**FER:** Frame Error Rate

**MTU:** Maximum Transmission Unit

**CPU:** Central Processing Unit

**LDPC:** Low Density Parity Check

**STBC:** Space–time block coding

**TxBF:** Transmit Beamforming

**MCS:** Modulating and Coding Scheme

## Palabras clave

---

Rendimiento, Nttcp, Iperf, Throughput, Jitter, Overhead, Linux, Wi-Fi, entorno doméstico, medidas, 802.11n, Retransmisión, Agregación de tramas y Mecanismo de transmisión (Rate adaptation).

# Capítulo 1. Introducción

---

## 1.1 Presentación y objetivos

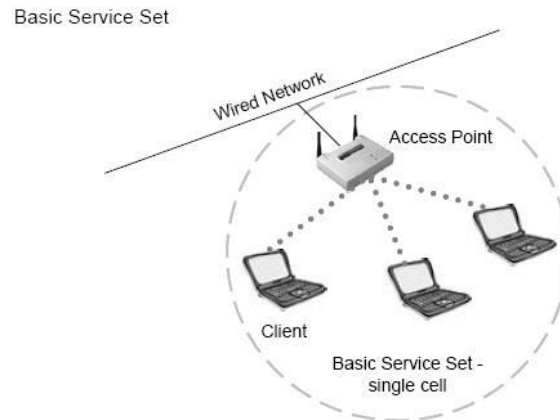
Desde de la aparición de las redes Wi-Fi en los años noventa, la tecnología Wi-Fi ha ido mejorando, aumentando su velocidad de transmisión a través de una serie de innovaciones, siendo las mejoras graduales con el paso del tiempo. Pero lo que hace más importante a la tecnología Wi-Fi es el alto grado de penetración que tiene esta tecnología y cada vez mayor en nuestra sociedad, como puede ser el empleo de tecnología Wi-Fi en teléfonos móviles y portátiles. Se puede decir que la tecnología Wi-Fi se ha convertido en un elemento imprescindible en nuestras vidas y no solo que la tecnología Wi-Fi se estanque, sino que vaya mejorando con el tiempo.

El 11 de septiembre fue ratificado por la organización IEEE el estándar 802.11n. Este estándar produjo una mejora significativa frente a sus estándares predecesores como por ejemplo de un incremento de la velocidad de transmisión desde los 54 Mbps hasta los 600 Mbps. Las dos grandes innovaciones de este estándar consisten en la incorporación de dos nuevas tecnologías. La primera de ellas es la tecnología MIMO que permite enviar y recibir información por múltiples antenas de manera simultánea. Gracias a esto se consigue dos características importantes como son la diversidad espacial y la diversidad de antenas. La segunda tecnología que incorpora el estándar 802.11n es el Channel Bonding, que permite canales de 40 MHz de anchura, gracias a la unión de dos canales de 20 MHz adyacentes. La unión de canales permite incrementar la cantidad de datos que pueden ser transmitidos. Utilizar conjuntamente una arquitectura MIMO con canales de mayor ancho de banda, ofrece la oportunidad de crear sistemas muy poderosos y rentables para incrementar la velocidad de transmisión de la capa física.

El objetivo de este proyecto será comprobar la calidad de estas conexiones Wi-Fi del estándar 802.11n en un entorno doméstico. Para ello se realizarán una serie de medidas utilizando principalmente el programa “nttcp”, con el cual usando un flujo de tráfico UDP se podrá comprobar cuantos paquetes se pierden, la velocidad de transmisión media y el tiempo que dura. A su vez, se capturará ese flujo con el programa Wireshark y se visualizará todos los datos importantes como los que ofrece el beacon de la red Wi-Fi en la que se van a realizar las medidas, que indicará las características de la red Wi-Fi como por ejemplo en que canal transmite o que características implementa, y el flujo de tráfico UDP creado con “nttcp” para poder ver las retransmisiones de tramas que se producen, la velocidad de transmisión y la potencia de señal recibida de cada ráfaga de tramas. Todos estos datos se extraerán con una serie de scripts escritos por el autor de este documento y se representarán gráficamente para una mejor comprensión de los datos.

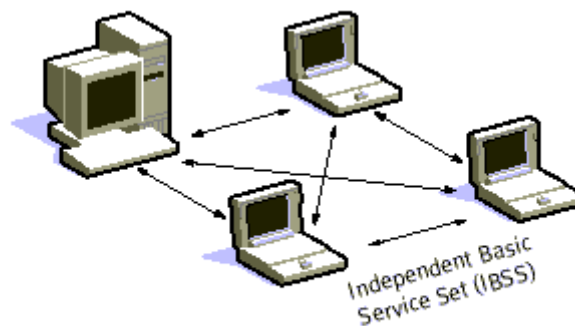
Para la realización de las medidas se trabajará en modo infraestructura, es decir, habrá un enrutador/punto de acceso Wi-Fi que será el router de mi hogar y dos ordenadores que harán de transmisor y de receptor. Cada ordenador se conectará a la red con un adaptador Wi-Fi USB. Al trabajar en modo infraestructura los dispositivos electrónicos en una misma red Wi-Fi se comunicarán entre si a través del router, por lo que habrá dos flujos de datos que teóricamente transmiten la misma información. El primer enlace será el enlace de subida, el flujo de tráfico

que se dirige desde el transmisor al router, y el otro flujo será el de baja, en el cual el flujo de tráfico va desde el router hasta el receptor.



*Figura 1.1. Red Wi-Fi en modo Infraestructura*

También se realizará otro conjunto de medidas, pero esta vez empleando la estructura de red Ad-hoc, en el cual las estaciones (clientes de la red) se comunican directamente sin la necesidad de un punto de acceso como puede ser un router. En este caso como la conexión es directa habrá un único flujo de datos.



*Figura 1.2. Red Wi-Fi en modo Ad-hoc*

Además, cabe destacar que se realizarán varias medidas en varios escenarios. Estos escenarios serán cuatro y se diferenciarán por la posición del transmisor (cerca o lejos) y por el horario de la realización de las medidas (tarde o mañana).

Por último, resaltar que en este trabajo el canal de la red Wi-Fi tendrá una anchura de 20 MHz y una velocidad de transmisión máxima de 130 Mbps ya que las medidas se realizaran con las características básicas del estándar 802.11n. La red Wi-Fi empleada en el modo Ad-hoc tendrá las mismas características que en el modo infraestructura.



## 1.2 Estructura de la memoria

El presente documento se estructura en 6 capítulos más dispuestos de la siguiente manera:

- En el capítulo 2 se explicarán una serie de aspectos teóricos relacionados con las redes Wi-Fi y las mejoras que aporta las redes Wi-Fi 802.11n.
- En el capítulo 3 se detallará todos los elementos que participarán en la realización de las medidas y los distintos tipos de escenarios que habrá, teniendo en cuenta la posición del transmisor y el horario de la realización de las medidas.
- En el capítulo 4 se detallará todos los programas usados y su instalación, lenguajes de programación y modificaciones en los elementos que participarán en la realización de las medidas para que estas medidas se puedan realizar.
- En el capítulo 5 se explicará cómo se realizan las medidas usando los programas nttcp e iperf, el filtrado y extracción de datos utilizando el programa Wireshark y un programa .c y finalmente la representación gráfica de esos datos usando un script Matlab.
- En el capítulo 6 se analizarán y representarán los resultados conseguidos para cada escenario seguidos de un comentario sobre estos.
- Por último, en el séptimo y último capítulo, se expondrán las conclusiones obtenidas de los resultados y posibles líneas de trabajo futuro.

## Capítulo 2. Conocimientos previos

---

En este apartado tratará de aspectos técnicos relacionados con el estándar 802.11, concretamente su evolución a lo largo del tiempo hasta llegar al estándar 802.11n sobre el cual se pondrá un especial énfasis, se detallarán sus características y sus mejoras respecto a estándares anteriores.

### 2.1 Estándar 802.11-1997

Este estándar es la versión original del estándar 802.11 publicada en 1997 por la IEEE (Instituto de ingenieros eléctricos y electrónicos en castellano). Este estándar especifica dos tasas binarias “teóricas” para la transmisión de datos que son 1 Mbps y 2 Mbps en la banda de frecuencia de 2,4 GHz. La anchura de cada canal es de 22 MHz. Se define a las tasas binarias como “teóricas” ya que buena parte de esta velocidad de transmisión teórica se empleará en la codificación para mejorar la calidad de transmisión cuando hubiera interferencias de cualquier tipo en el canal. Este problema se solucionará en los estándares posteriores a este. Para la transmisión de datos en la capa física se puede emplear tres métodos: infrarrojos, espectro ensanchado por salto de frecuencia y espectro ensanchado en secuencia directa. En este estándar también define el protocolo “múltiple acceso por detección de portadora evitando colisiones” (CSMA/CA) como método de acceso. Actualmente este estándar está obsoleto.

### 2.2 Estándar 802.11b

El estándar 802.11b fue aprobado en 1999. El estándar 802.11b para resumir es una versión más extendida del primer estándar 802.11, permitiendo una velocidad de transmisión máxima de 11 Mbps, trabajando en la misma banda de los 2,4 GHz, empleando el mismo método de acceso CSMA/CA y con la misma anchura de 22 MHz para cada canal. La tecnología usada en la capa física será también el ensanchamiento de espectro por secuencia directa. Para cada velocidad de transmisión se empleará una modulación digital diferente: para 1 Mbps → DBPSK, para 2 Mbps → DQPSK y finalmente tanto para 5,5 Mbps como para 11 Mbps → CCK.

Es gracias a este estándar que comenzó el auge de las redes Wi-Fi. La gran mejora de rendimiento respecto al estándar original y su bajo precio provocó una rápida aceptación entre el público y la hizo adecuada para entornos domésticos. Este estándar sigue estando vigente pero cada vez tiene menos importancia debido a la salida de estándares que mejoran las características de este.

### 2.3 Estándar 802.11a

El estándar 802.11a fue aprobado en 1999. Este estándar especifica que se trabaja en la banda de los 5 GHz y se basa en OFDM. La técnica de transmisión OFDM consiste básicamente en dividir la banda en canales ortogonales y que cada uno se use para transmitir una portadora. La decisión de trabajar en la banda de los 5 GHz viene dada a la excesiva ocupación que sufre la banda de los 2,4 GHz. El problema que se origina al trabajar en esta banda de los 5 GHz es que las longitudes de onda son mucho menores y por ello son absorbidas mucho más fácilmente por

los obstáculos. Como ejemplo, 802.11b tiene un rango mucho más grande para velocidades bajas. Los canales se numeran cada 5 MHz y tienen una anchura de 20 MHz.

Se utilizan 48 subportadoras y se trabaja a una velocidad de 250000 símbolos por segundo. Para terminar su velocidad máxima de transmisión es 54 Mbps, permitiendo 8 diferentes velocidades binarias de trabajo (6, 9, 12, 18, 24, 36, 48, 54 Mbps) utilizando diversas modulaciones (BPSK, QPSK, 16-QAM y 64-QAM). Este estándar sigue vigente.

## **2.4 Estándar 802.11g**

El estándar 802.11g fue aprobado en 2003 y es una evolución directa del estándar 802.11b. Al ser una extensión de 802.11b compartirá características como por ejemplo trabajar en la banda de los 2,4 GHz. Los canales pueden tener una anchura de 20 MHz o de 22 MHz (como 802.11b). Este estándar podrá usar como en 802.11b en la capa física para la transmisión de datos el ensanchamiento de espectro por secuencia directa y la novedad sería el empleo de la técnica de transmisión OFDM. Este estándar es totalmente compatible con 802.11b pudiendo un dispositivo con 802.11g conectarse a un punto de acceso que usa 802.11b y al revés. La velocidad de transmisión máxima es 54 Mbps, pudiendo variar la velocidad de trabajo dependiendo de la modulación digital empleada. Este estándar sigue estando vigente.

## **2.5 Estándar 802.11n**

Antes de comenzar con este estándar sobre el cual se realizará todo el trabajo, recalcar que además de lo definido en los estándares por su creador, la IEEE, hay que tener en cuenta las normas creadas por las autoridades de regulación de cada país para los protocolos inalámbricos. En el caso de España, está prohibido usar el canal 14 en la banda de los 2,4 GHz y la potencia de transmisión máxima es de 100 mW.

El estándar 802.11n fue publicado en octubre de 2009 concretamente. Este estándar es una versión extendida del estándar 802.11g por lo que será compatible con los dispositivos que empleen los estándares 802.11b y 802.11g con la adición de los dispositivos que utilicen el estándar 802.11<sup>a</sup>. El estándar 802.11n puede trabajar tanto en la banda de los 2,4 GHz como de los 5 GHz. La anchura de los canales puede ser tanto de 20 MHz como de 40 MHz.

La inclusión de canales de 40 MHz es una nueva ventaja incorporada en este estándar y se consigue con el empleo de la tecnología Channel Bonding mencionada en la Introducción, que consiste en la unión de dos canales adyacentes que no solapan de 20 MHz, ya sea de la banda de frecuencias de los 5 GHz o de los 2,4 GHz, para la creación de un canal de anchura de 40 MHz, permitiendo incrementar la cantidad de datos transmitidos. Aunque esta tecnología es una novedad en este estándar, no está obligada su implementación, es una característica adicional que la puede incorporar o no el punto de acceso que use el estándar 802.11n.

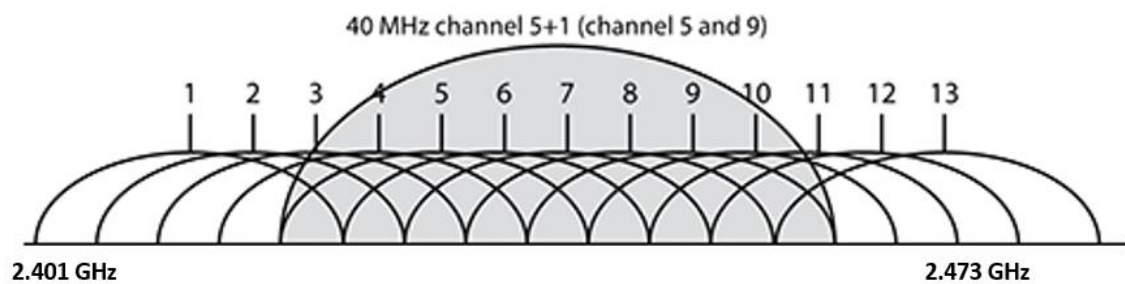


Figura 2.1. Channel bonding

La principal novedad del estándar 802.11n es la técnica de transmisión MIMO, mencionada en la introducción. La tecnología MIMO consiste en transmitir y recibir flujos de información por múltiples antenas. Estos flujos de datos cada uno se denominan “spatial stream” y el punto de acceso deberá como mínimo soportar dos spatial streams, siendo el uso de 3 y 4 spatial streams opcional. Las principales ventajas de MIMO son la diversidad de antenas para la transmisión de datos y la diversidad espacial, que, aunque ya se podía emplear esta última característica en estándares anteriores, no se empleaba simultáneamente para conseguir un multiplexado espacial real como el que ofrece el estándar 802.11n. El beneficio de MIMO es menor cuando hay una clara línea de vista, ya que las señales multiruta llegan más tarde que la señal de línea de visión, pero debido a la diversidad de señales multiruta que proporciona la tecnología MIMO permite al receptor recuperar los mensajes de la señal, lo cual hace esta tecnología muy apta para entornos interiores con obstáculos.

Otra ventaja que proporciona MIMO para la transmisión de información es SDM (Multiplexado de división espacial). Consiste en multiplexar espacialmente flujos de datos independientes simultáneamente. SDM puede mejorar el rendimiento de la transmisión conforme el número de spatial streams es incrementado. Cada spatial stream (flujo espacial en español) requiere una antena discreta tanto en el transmisor como en el receptor. El único problema de tener un gran número de spatial streams es que cada spatial stream requiere una cadena de radio frecuencia y un convertidor de analógico a digital para cada antena lo cual incrementa significativamente el coste.

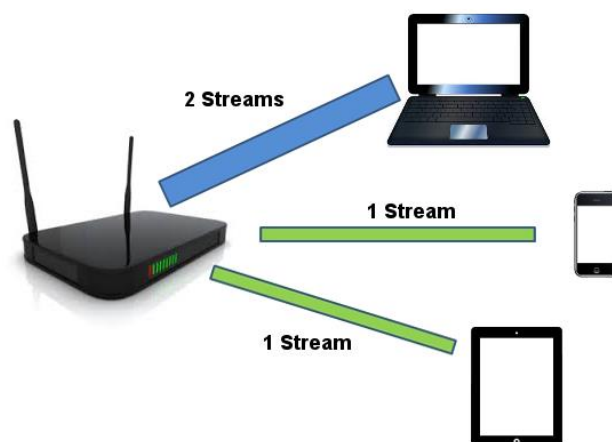


Figura 2.2. MIMO

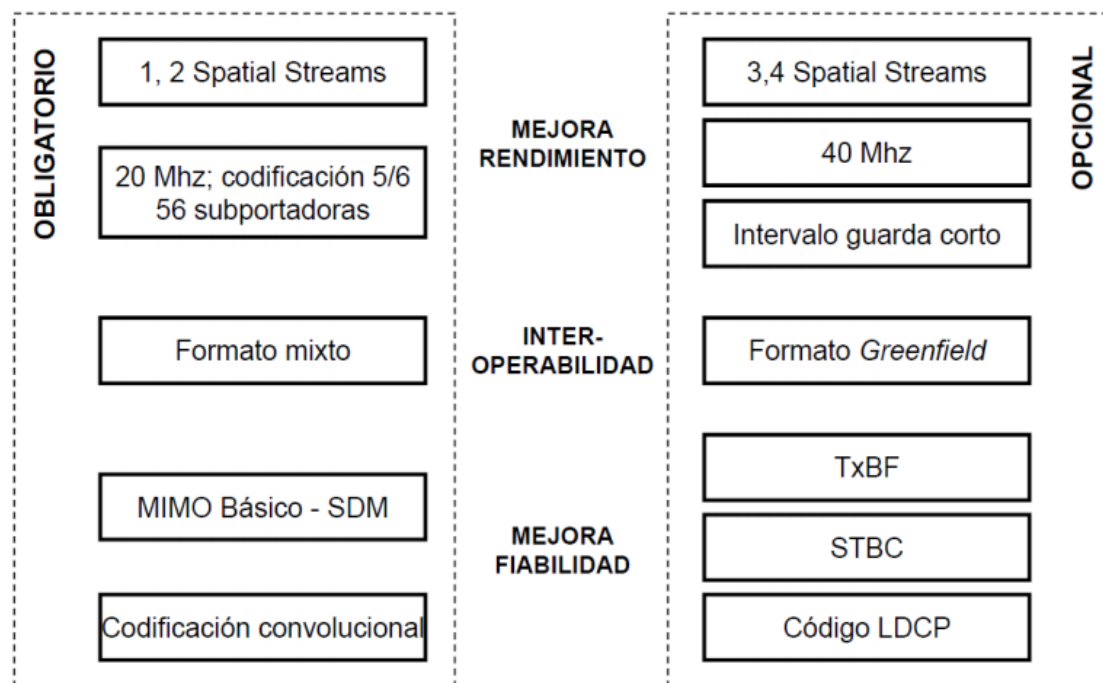
La velocidad de transmisión en el estándar 802.11n como ya se menciona es 600 Mbps, pero esto depende de que se den las características y condiciones adecuadas, como por ejemplo un canal de anchura de 40 MHz, un intervalo de guarda corto (400 ns frente a los 800 ns del intervalo de guarda normal), el número máximo de spatial streams, 4, el tipo de modulación digital usada (64-QAM para 600 Mbps) y el tipo de codificación empleada ( $R=5/6$  para 600 Mbps). Variando cualquiera de estos valores la velocidad de transmisión cambiará, por lo que el estándar 802.11n proporciona un amplio rango de velocidades de transmisión. A continuación, se pondrá una lista de las velocidades de transmisión posibles según el número de spatial streams, anchura del canal, tipo de codificación, tipo de intervalo de guarda y la modulación digital empleada. Para destacar, todos los puntos de acceso que emplean el estándar 802.11n deberían tener una velocidad de transmisión máxima de 130 Mbps o superior.

*Tabla 2.1. Lista velocidades de transmisión del estándar 802.11n*

Index MCS	Modul.	Codif.	Spatial Streams	Canales 20 Mhz		Canales 40 Mhz	
				800 ns	400 ns	800 ns	400 ns
0	BPSK	1/2	1	6.5	7.2	13.5	15.0
1	QPSK	1/2	1	13.0	14.4	27.0	30.0
2	QPSK	3/4	1	19.5	21.7	40.5	45.0
3	16-QAM	1/2	1	26.0	28.9	54.0	60.0
4	16-QAM	3/4	1	39.0	43.3	81.0	90.0
5	64-QAM	2/3	1	52.0	57.8	108.0	120.0
6	64-QAM	3/4	1	58.5	65.0	121.5	135.0
7	64-QAM	5/6	1	65.0	72.2	135.0	150.0
8	BPSK	1/2	2	13.0	14.4	27.0	30.0
9	QPSK	1/2	2	26.0	28.9	54.0	60.0
10	QPSK	3/4	2	39.0	43.3	81.0	90.0
11	16-QAM	1/2	2	52.0	57.8	108.0	120.0
12	16-QAM	3/4	2	78.0	86.7	162.0	180.0
13	64-QAM	2/3	2	104.0	115.6	216.0	240.0
14	64-QAM	3/4	2	117.0	130.0	243.0	270.0
15	64-QAM	5/6	2	130.0	144.4	270.0	300.0
16	BPSK	1/2	3	19.5	21.7	40.5	45.0
17	QPSK	1/2	3	39.0	43.3	81.0	90.0
18	QPSK	3/4	3	58.5	65.0	121.5	135.0
19	16-QAM	1/2	3	78.0	86.7	162.0	180.0
20	16-QAM	3/4	3	117.0	130.0	243.0	270.0
21	64-QAM	2/3	3	156.0	173.3	324.0	360.0
22	64-QAM	3/4	3	175.5	195.0	364.5	405.0
23	64-QAM	5/6	3	195.0	216.7	405.0	450.0
24	BPSK	1/2	4	26.0	28.9	54.0	60.0
25	QPSK	1/2	4	52.0	57.8	108.0	120.0
26	QPSK	3/4	4	78.0	86.7	162.0	180.0
27	16-QAM	1/2	4	104.0	115.6	216.0	240.0
28	16-QAM	3/4	4	156.0	173.3	324.0	360.0
29	64-QAM	2/3	4	208.0	231.1	432.0	480.0
30	64-QAM	3/4	4	234.0	260.0	486.0	540.0
31	64-QAM	5/6	4	260.0	288.9	540.0	600.0

Otro beneficio del estándar 802.11n es su compatibilidad con los estándares 802.11g y 802.11a. Esta compatibilidad se conseguirá añadiendo un segundo preámbulo para MIMO. En canales de 40 MHz se repetirá este preámbulo en cada canal de 20 MHz que lo compone el canal de 40 MHz. Este método se denomina formato mixto. En el caso de que no haya dispositivos que empleen los antiguos estándares 802.11g y 802.11a, se podrá emplear el formato GreenField, que define un preámbulo exclusivo para 802.11n y mejora sus prestaciones. El formato Greenfield es una característica opcional y no hace falta que todos los dispositivos lo incorporen.

Debido a la compatibilidad con estándares anteriores, el estándar 802.11n seguirá empleando la tecnología de transmisión OFDM aumentando la codificación convolucional máxima a 5/6, añadiendo 4 subportadoras y una característica que es opcional será el empleo de un intervalo de guarda más corto, concretamente de 400 ns, frente al intervalo de guarda anterior que era de 800 ns.



*Figura 2.3. Características obligatorias frente a las características opcionales del estándar 802.11n*

A continuación, se mencionará una serie de características opcionales que ofrece el estándar 802.11n para mejorar la fiabilidad. El primero de ellos es el Transmit Beamforming (TxBF) que básicamente es una característica que permite al punto de acceso concentrar efectivamente la señal en el lugar del usuario de la red Wi-Fi. Para ello es necesario conocer la situación del canal y actualmente es poco usado.

La segunda mejora opcional es el Space-Time Block Coding (STBC), que es una codificación que incorpora redundancia para mejorar la fiabilidad. A través de varios spatial streams se mandarán diferentes versiones codificadas de la señal.

La tercera y última mejora opcional es la codificación Low Density Parity Check (LDPC) que mejora las prestaciones del código convolucional tradicional. [1]

## 2.6 Agregación de tramas en el estándar 802.11n

La agregación de tramas es una característica diseñada en el estándar 802.11e y que es por primera vez utilizada en el estándar 802.11n y que consiste en la transmisión de dos o más tramas en una sola transmisión. Esto mejora drásticamente el rendimiento y eficiencia de las transmisiones de datos debido al ahorro relacionado con la sobrecarga de múltiples transmisiones, como puede ser el empleo de una misma cabecera para un conjunto de mensajes. Para la confirmación de estas ráfagas de tramas en el caso de A-MPDU ya no se empleará la trama ACK sino la nueva trama BLOCK ACK, que confirma si se han recibido todas las tramas de una ráfaga correctamente o especifica si alguna no ha llegado correctamente. El ACK solo servía para confirmar tramas MPDU una por una mientras que el BLOCK ACK permitía la confirmación de una ráfaga de tramas MPDU, aumentando la eficiencia en la capa MAC. Para la agregación de tramas hay dos técnicas distintas empleadas: A-MPDU y A-MSDU.

### 2.6.1 A-MSDU

A-MSDU consiste en la agrupación de MSDUs (MAC service data unit), que son las unidades de servicio de datos que se recogen de la sub-capa LLC y que está sobre la sub-capa MAC. En este caso todas las tramas MSDU de una misma ráfaga A-MSDU comparten la cabecera MAC. El objetivo de A-MSDU es permitir múltiples MSDUs ser enviadas a un mismo receptor concatenadas como una MPDU (MAC Protocol data unit). Una MPDU consiste en el conjunto de datos recogidos de la sub-capa MAC. Para poder realizar una agregación A-MSDU es necesario la dirección destino (DA) y la dirección del emisor (SA) deben coincidir a las direcciones transmisor (TA) y receptor (RA) de la cabecera MAC, razón por la cual A-MSDU no puede ser empleado para tráfico Broadcast o Multicast.

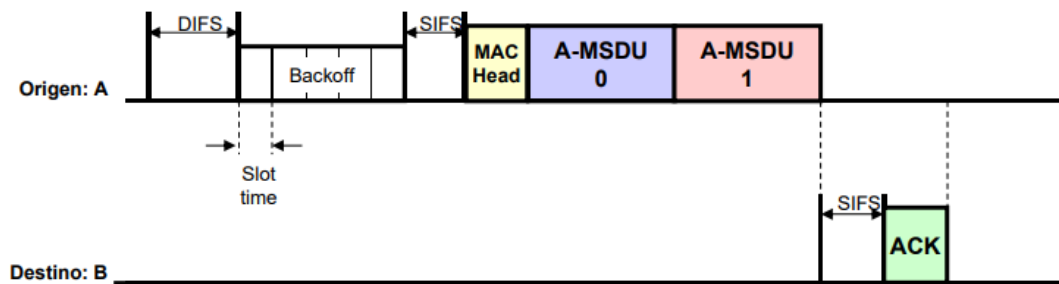
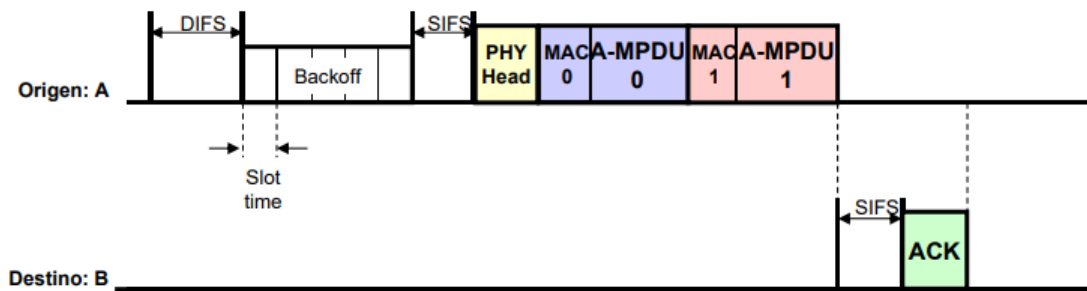


Figura 2.4. Funcionamiento agregación A-MSDU

### 2.6.2 A-MPDU

El objetivo de la A-MPDU es juntar varias tramas MAC (MPDU) en una transmisión en la que comparten la cabecera física, teniendo cada trama MAC (MPDU) su cabecera MAC no como en A-MSDU que todas las tramas MSDU compartían una cabecera MAC. Aunque la ráfaga A-MPDU se transmita del mismo sitio, puede tener destinos diferentes cada trama MAC. Este método suele ser menos eficiente que el método A-MSDU, pero en entornos interiores con obstáculos como puede ser un entorno doméstico tiene una mayor eficiencia ya que gracias al BLOCK ACK permite la retransmisión selectiva de las tramas erróneas y no de toda la ráfaga A-

MPDU, no como ocurre en A-MSDU. En este trabajo las transmisiones de datos se realizarán empleando A-MPDUs. El número de tramas MAC que contendrá una A-MPDU vendrá definido por los bytes que podrá transmitir como máximo en una A-MPDU y el tiempo máximo que debe durar una A-MPDU. En este caso nunca habrá limitación por exceso de tamaño, pero sí de tiempo. Se podrá comprobar que dependiendo de la velocidad de transmisión de la A-MPDU contendrá más trama MAC (alta velocidad de transmisión) o no (baja velocidad de transmisión). [2]



*Figura 2.5. Funcionamiento agregación A-MPDU*



## Capítulo 3. Descripción de la plataforma de caracterización

---

En este apartado se definirá el escenario en el cual se van a realizar las medidas, los actores que participarán en la realización de las medidas ya sea como transmisión, retransmisión o recepción de datos y por último la calidad del medio en el cual se va a realizar las medidas, es decir, si hay interferencias por la presencia de más redes Wi-Fi.

### 3.1 Elementos

En este subapartado se detallará los elementos que participan en la realización de las medidas con sus especificaciones y características. Los elementos que se detallarán son el portátil que hará el papel de transmisor, el router que reenviará los paquetes al ordenador receptor y por último el ordenador fijo que será el receptor. Además, para la conexión a la red Wi-Fi por parte de los ordenadores para la realización de las medidas se utilizarán unos adaptadores USB Wi-Fi mencionados en la introducción.

#### 3.1.1 Transmisor

El ordenador portátil que se encargará de la transmisión y captura de paquetes tiene las siguientes especificaciones:

- Placa base: Aspire A315-21
- Procesador: AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G (2 CPUs), 3,1 GHz
- RAM instalada: 12 GB

#### 3.1.2 Router

En el modo infraestructura será necesario utilizar un Router que hará de intermediario en el flujo de datos. El router que se utilizará será de la marca Arcadyan modelo prv3399b-b-lt cuya descripción del hardware es la siguiente:

- Procesador MIPS-32 Dual Core 600 MHz
- Memoria NAND FLASH de 128MB y DDRAM de 256MB
- Modo operacional: FTTH con ONT integrada (conexión directa a PTR-O)
- 4 puertos LAN Gigabit Ethernet (GE) RJ45
- 2 puertos (FXS) RJ11 para Telefonía
- 1 puertos USB 2.0 maestro (tipo A) de alta potencia (1A)
- Wi-Fi Doble Banda 11ac MU-MIMO 4x4, 11n MIMO 2x2



*Figura 3.1. Router Arcadyan prv3399b-b-It*

El router permitirá crear dos tipos distintos de red Wi-Fi que se diferencian por la banda en la que trabajan. El primer tipo de red se encuentra en la banda de los 2,4 GHz y es la que se empleará en la realización de las medidas, en su vertiente 802.11n. Las especificaciones de la red Wi-Fi en la banda 2,4 GHz es la siguiente:

- 802.11b/g/n MIMO 2x2 (2 antenas) hasta 300 Mbps
- Ancho de canales hasta los 40 MHz. En nuestro caso el canal tendrá una anchura de 20 MHz.
- El segundo tipo de red Wi-Fi se centra en la banda de los 5 GHz y sus especificaciones son las siguientes:
- 802.11ac/n/a MIMO 4x4 (4 antenas) hasta 1700 Mbps
- Ancho de canal de hasta 80 MHz

Esta última red no se empleará, aunque acepte conexiones tipo 802.11n, ya que los adaptadores Wi-Fi USB no trabajan en la banda de los 5 GHz, solo trabajan en la banda de 2,4 GHz, lo cual solo deja una elección posible a la hora de elegir tipo de red Wi-Fi.

### **3.1.3 Receptor**

El ordenador fijo que se encargará de la recepción de paquetes tiene las siguientes especificaciones:

- Placa base: MSI modelo MS-7A72
- Procesador: Intel® Core™ i7-7700 de 3,6 GHz
- RAM instalada: 16 GB

### **3.1.4 Adaptador USB Wi-Fi**

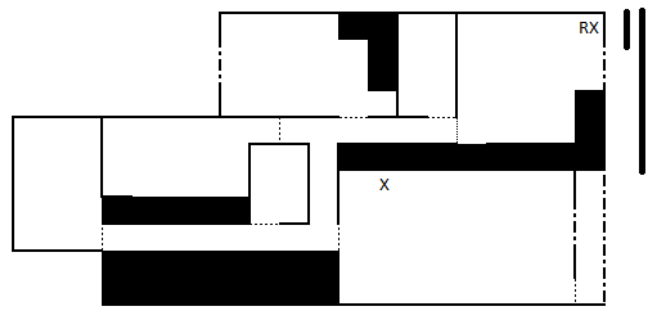
Para la realización de medidas se utilizará un adaptador Wi-Fi USB para conectarse a la red Wi-Fi ya sea tanto para la transmisión de paquetes, recepción de paquetes y monitorización del tráfico. El adaptador Wi-Fi USB utilizado TP-LINK modelo TL-WN821N (firmware v3) con un chipset Atheros AR9002U-2NG. Además, contendrá dos tipos de sistema en el chip (SOC = System on a Chip), el AR7010 y el AR9287, ambos modelos pertenecientes a Atheros. Este adaptador inalámbrico puede llegar a alcanzar los 300 Mbps de velocidad en una red Wi-Fi tipo 802.11n, que es en la cual se van a realizar las medidas, y solo cumple su función en la banda de frecuencia de los 2,4 GHz.



*Figura 3.2. Adaptador inalámbrico TP-LINK modelo TL-WN821N*

### 3.2 Escenario

El escenario en el cual se van a realizar las pruebas corresponde a mi casa, es decir, un entorno doméstico. El plano de mi casa es el siguiente:



*Figura 3.3. Plano aproximado del entorno*

Lo primero que haré será explicar el plano. Lo primero es la escala. Las líneas de la derecha del plano tienen un valor de 1 metro, 6 metros y 9 metros respectivamente. Con esto se podrá calcular todas las longitudes de las paredes si fuera requerido y saldrían valores exactos ya que hay que recordar que es un plano aproximado.

Cada tipo de borde representa una cosa diferente. Los bordes negros finos normales representan paredes normales, mientras que los bordes gruesos (que ocupan un metro de grosor en el plano) representan paredes gruesas u obstáculos. Las líneas de puntos representarían las puertas de las habitaciones, es decir, la entrada a ellas. Por último, estarán las líneas segmentadas divididas por puntos que corresponderán a las ventanas.

En el plano aparecen dos elementos, “RX” y “X”. Estos elementos son el ordenador fijo que se encargará de recibir las tramas y el router respectivamente. Se representan en el plano porque son elementos fijos a la hora de realización de las medidas ya que no se van a mover en ningún momento.

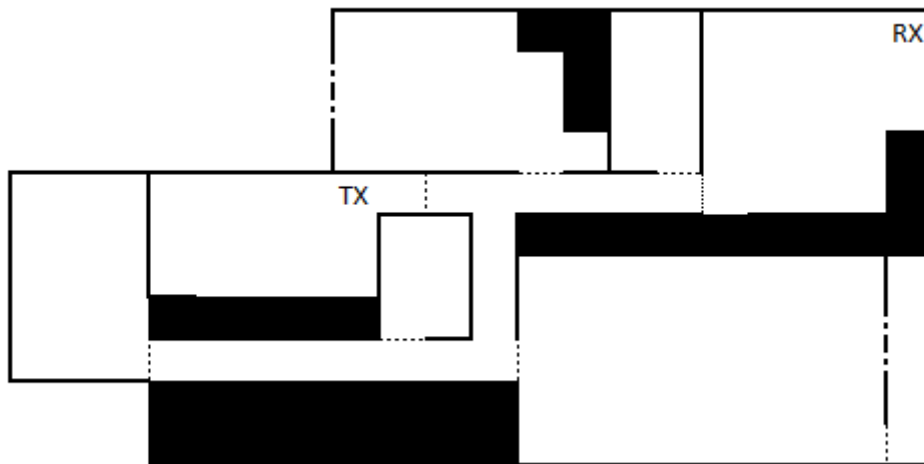
A la hora de realizar las medidas en modo infraestructura habrá dos tipos de escenarios teniendo en cuenta donde se coloca el ordenador portátil transmisor de paquetes. En el primer escenario se colocará el transmisor al lado del router, mientras que en el segundo escenario el transmisor estará al lado del receptor. Es decir, en todas las medidas el enlace de bajada (Router -> RX) será siempre el mismo, solo variará el enlace de subida (TX->Router).

Cabe destacar que sabiendo las longitudes se podrá calcular la distancia entre "X" y "RX". Definiendo un triángulo rectángulo en el que la hipotenusa represente la distancia entre "X" y "RX" y en el que se sepa la longitud de los catetos, aplicando el teorema de Pitágoras se podrá obtener el valor de esa hipotenusa:

$$\text{Distancia } X - RX = \sqrt{8^2 + 7^2} = 10,6 \text{ metros aprox.}$$

Este valor sería para una línea recta entre ambos puntos sin tener en cuenta obstáculos.

Para la realización de las medidas con una red Wi-Fi en modo Ad-hoc, también se empleará dos escenarios: el primero que consiste el transmisor y el receptor están al lado (recordar que el receptor estará siempre fijo) por lo que la conexión será la mejor posible y otra que será en la que el receptor y transmisor estén alejados, como se verá en la imagen inferior, siendo la conexión peor:



*Figura 3.4. Escenario con red Wi-Fi en modo Ad-hoc*

Este cambio en el escenario en las medidas realizadas cuando el transmisor y receptor están alejados empleando una red Wi-Fi en modo Ad-hoc es debido a que ambos ordenadores no se pueden interconectar entre sí si uno de ellos se coloca al lado del router debido a las limitaciones de la red Ad-hoc y a los obstáculos.

En este caso la distancia entre el transmisor y el receptor es de aproximadamente 14 metros en línea recta.

### 3.3 Tráfico interferente en el escenario

En este apartado se detallará como estará de congestionado el canal de la red Wi-Fi, que en mi caso es el 1, debido a las interferencias de otras redes Wi-Fi mientras se realizan las medidas.

A la hora de realizar las medidas habrá dos escenarios diferentes esta vez teniendo en cuenta el impacto de las interferencias de otras redes Wi-Fi, siendo el primer escenario durante la mañana y el segundo escenario en la tarde. ¿Por qué se realizan medidas por la tarde y por la mañana? Debido a que el número de redes Wi-Fi es mayor por la tarde mientras que por la mañana hay menos redes Wi-Fi activas.

Para ver las redes Wi-Fi de nuestro entorno se habrá que descargar e instalar el programa “inSSIDer Home”, se conectará el adaptador Wi-Fi USB al ordenador para que el programa pueda funcionar, y una vez abierto el programa, se entrará a un interfaz en el cual se podrá ver una lista de redes Wi-Fi que se irá incrementando a medida que se detecten más redes Wi-Fi. En esta lista aparecerá indicado el nombre de cada red Wi-Fi, su dirección MAC, su tipo de seguridad, que tipo de red Wi-Fi 802.11 es (b, g, n), su canal y la potencia de señal recibida.

En la parte inferior, habrá dos gráficos donde se representarán las redes Wi-Fi, según su canal en el eje x y la potencia de señal recibida en el eje y. Un gráfico será para la banda de 2,4 GHz que será el único con información, ya que como se mencionó anteriormente el adaptador Wi-Fi USB sólo trabaja en la banda de 2,4 GHz y sólo captará redes Wi-Fi de esta banda, y el otro gráfico será para la banda de 5 GHz y estará vacía.

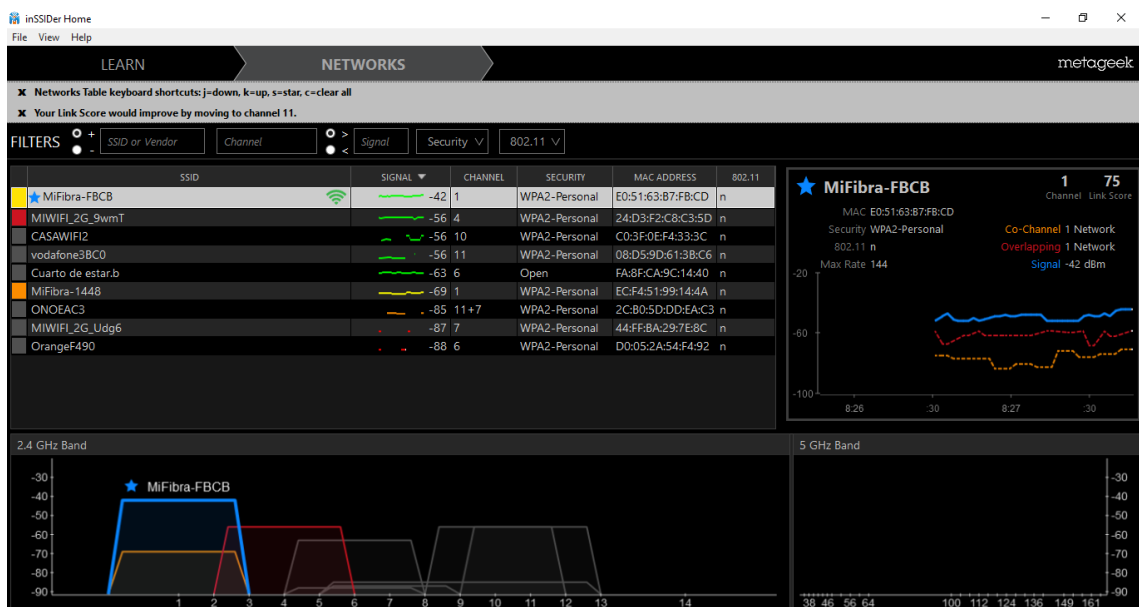


Figura 3.5. Interfaz programa inSSIDer Home

En la parte derecha, si se selecciona una red como en mi caso la red Wi-Fi de mi casa “MiFibra-FBCB”, nos aparecerá un gráfico indicándonos varios datos conocidos añadiendo uno nuevo que es la máxima velocidad que nos permite esa red Wi-Fi, en nuestro caso 144 Mbps. La

parte inferior del gráfico nos enseñará la evolución de la potencia de la señal recibida de la red que se ha seleccionado (azul), de las redes con las que comparte canal (naranja) y las redes que se superponen a la red seleccionada (rojo). El eje x representa el tiempo transcurrido y el eje y representa la potencia de señal recibida. En la esquina superior derecha nos indicará cuantas redes comparten canal con la red seleccionada (co-channel), cuantas se superponen (overlapped) y un indicador exacto de la potencia de señal recibida de nuestra red. Fijándose que, en la lista de redes mencionada antes en este mismo apartado, las redes que tengan una marca naranja serán las redes co-canal de nuestra red, las redes overlapped tendrán una marca roja y por último nuestra red una marca amarilla.

Después de realizar varias comprobaciones en días diferentes sobre las interferencias de otras redes Wi-Fi para los cuatro escenarios posibles en modo infraestructura teniendo en cuenta el horario y la posición del transmisor, siendo en el transmisor donde se observa estas posibles interferencias, estos son los resultados:

- Escenario 1 (transmisor al lado del router y por la mañana): la potencia de señal recibida para la red Wi-Fi en la cual se realizarán las medidas (MiFibra-FBCB) varía entre -42 y -50 dBm habiendo una red Wi-Fi co-canal y una red Wi-Fi overlapped.
- Escenario 2 (transmisor al lado del ordenador receptor (RX) y por la mañana): la potencia de señal recibida para la red Wi-Fi en la cual se realizarán las medidas (MiFibra-FBCB) varía entre -55 y -65 dBm habiendo entre 3 o 4 redes Wi-Fi co-canal y 1 o 2 redes Wi-Fi overlapped. (En este caso es más común que haya 3 redes co-canal y una red overlapped pero estos valores varían porque debido a que hay redes Wi-Fi lejanas a veces las captan intermitentemente)
- Escenario 3 (transmisor al lado del router y por la tarde): la potencia de señal recibida para la red Wi-Fi en la cual se realizarán las medidas (MiFibra-FBCB) varía entre -47 y -50 dBm habiendo una red Wi-Fi co-canal y una red Wi-Fi overlapped.
- Escenario 4 (transmisor al lado del ordenador receptor (RX) y por la tarde): la potencia de señal recibida para la red Wi-Fi en la cual se realizarán las medidas (MiFibra-FBCB) varía entre -55 y -68 dBm habiendo 5 redes Wi-Fi co-canal y 2 redes Wi-Fi overlapped.

Por lo tanto, lo que se podrá averiguar es que las interferencias del tráfico son mayores en los escenarios 2 y 4 con valores similares siendo ligeramente peor por la tarde. En los escenarios 1 y 3 son prácticamente idénticos, pasando igual que antes que por la tarde es ligeramente peor (a veces hay picos de mala calidad, pero casi imperceptible). Para las medidas de la red en modo Ad-hoc las interferencias con otras redes Wi-Fi obviamente serán las mismas que para la red en modo infraestructura. En este caso el escenario 1 representaría receptor y transmisor juntos por la mañana, el escenario 2 receptor y transmisor lejos por la mañana, el escenario 3 receptor y transmisor juntos por la tarde y el escenario 4 receptor y transmisor lejos por la tarde.

## Capítulo 4. Configuración de los equipos

Una vez definido el entorno y todos los agentes que entrarán en juego, se tendrá que realizar una serie de modificaciones e instalaciones de programas en los ordenadores transmisor y receptor para poder realizar las medidas y que estas sean correctas. Este apartado se centrará en explicar que programas se usarán, su instalación dependiendo del sistema operativo, los drivers que los adaptadores USB Wi-Fi utilizan y la creación de una partición en los ordenadores para la posterior instalación de un sistema operativo Linux, en este caso Ubuntu.

### 4.1. Crear partición e instalación Ubuntu

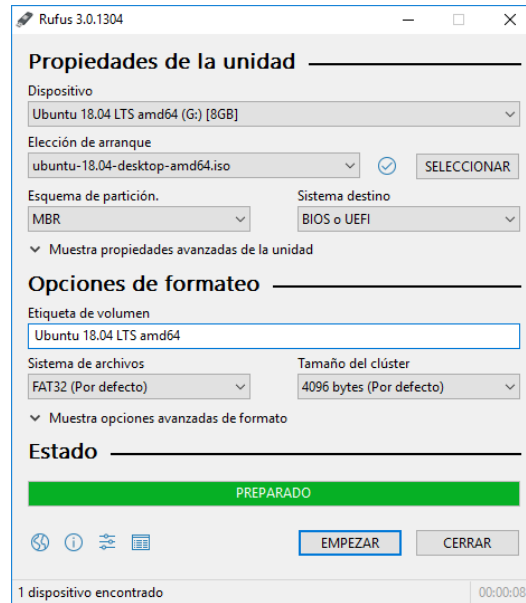
Para realizar las medidas se empleará el sistema operativo Ubuntu tanto en el ordenador fijo como en el ordenador portátil. Para ello antes de nada se creará una partición en cada ordenador con espacio suficiente para albergar el sistema operativo Ubuntu. En mi caso, debido a problemas de espacio en el ordenador fijo solo lo he podido asignar a la partición 10 GB por lo que va justo de memoria mientras que en el ordenador portátil asigne unos 50 GB. La versión de Ubuntu utilizada es la 4.15.0-38-generic de 64 bit. Para crear la partición en Windows 10 como es mi caso, simplemente buscando en el menú de inicio “Crear y formatear particiones del disco duro”. Una vez ahí, se elegirá la unidad de la que se quiere ceder espacio, en mi caso la unidad C, y se seleccionará la opción “reducir volumen”, y se liberará el espacio necesario que ya he mencionado anteriormente. El espacio liberado se tendrá que dejar como no asignado.



Disco 0 Básico 238,46 GB En pantalla	100 MB Correcto (Partición de sistema)	Acer (C:) 187,36 GB NTFS Correcto (Arranque, Archivo de paginación, Volcado, Partición primaria)	50,00 GB Correcto (Partición primaria)
---	---	--	---

*Figura 4.1. Partición de 50 GB del ordenador portátil*

Una vez creada la partición, se tendrá que ripear un USB, que es el método que utilizare para instalar Ubuntu en los ordenadores. Para ello utilizare el programa Rufus el cual es un programa que no necesita instalación y es relativamente sencillo. Una vez abierto, como se verá en la imagen, solamente se tendrá que seleccionar el dispositivo USB a ripear, y la imagen .iso de Ubuntu la cual se habrá descargado previamente de la página web oficial de Ubuntu.



*Figura 4.2. Interfaz del programa Rufus*

Por último, solo queda introducir el USB al ordenador cuando este se encuentre apagado, encenderlo y pulsar antes de que aparezca el icono de Windows la tecla predeterminada para elegir desde que unidad iniciar el equipo. Una vez seleccionada la opción que no es Windows, nos saltará la aplicación de instalación de Ubuntu, a través de la cual se configurarán algunas características del sistema operativo y se seleccionará la partición que se ha preparado anteriormente para el sistema operativo para que lo ocupe. Una vez terminado la instalación, ya se tendrá nuestro sistema operativo Ubuntu en perfectas condiciones para comenzar el trabajo.

## 4.2 Herramientas de medida

Como herramientas para generar tráfico y caracterizar el canal en mi trabajo se han empleado el programa Nttcp y el programa Iperf. Estas de herramientas software de medida, Ntcccp e Iperf, estarán ambas instalas en los dos ordenadores en el sistema operativo Linux para poder realizarse las medidas.

### 4.2.1 Nttcp

Nttcp (new test TCP program) mide el ratio de transferencia en una conexión UDP o TCP, además de otros parámetros. En este trabajo Nttcp será la principal herramienta de medida mientras que Iperf se usará ocasionalmente. Nttcp cada vez comience una transferencia de datos comenzará con la transmisión de una trama de tamaño muy pequeño (aproximadamente 127, varía según las cabeceras) que indicará el inicio de la transmisión y se volverá a utilizar esta pequeña trama para marcar el final de la transmisión.



### 4.2.2 Iperf

Iperf es una herramienta para medidas del máximo ancho de banda posible sobre redes IP. Soporta varios parámetros relacionados con el tiempo, buffer y protocolos (puede usar TCP y UDP, por ejemplo). Para cada prueba reporta el ancho de banda (solo en TCP), pérdidas y otros parámetros, como se verá más adelante. En este caso se utilizará el protocolo UDP.

### 4.3 Programas para análisis, obtención y representación de datos

Para pasos posteriores, como, por ejemplo, la representación de capturas, el filtrado de datos, la obtención de datos deseados o la creación de gráficas que representen esos datos, se tendrá que usar tres programas:

- El primero de ellos es el Wireshark. Wireshark es un analizador de protocolos y con él se podrá monitorizar el flujo de datos que se produzca y analizarlo. También se podrá utilizar algunas opciones de filtrado para que solo nos parezcan las tramas deseadas, como, por ejemplo, sólo tramas UDP.
- El segundo es Dev-C++. Este programa es un entorno de desarrollo integrado para programar en lenguaje C/C++. Utilizaremos Dev-C++ para crear el programa .C que se encargará de obtener los datos necesarios de la captura del flujo de datos.
- El último es Matlab. Este programa es un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio, el lenguaje M. Con este programa escribiremos un programa .M para que una vez obtenidos los datos extraídos del programa .C, el programa .M se encargue de representarlos en forma de gráficas de la manera más adecuada.



*Figura 4.3. Iconos de Wireshark, Dev-C++ y Matlab en este orden*

Cabe destacar que, aunque se haya instalado el sistema operativo Linux, solamente el Wireshark será utilizado en Linux para capturar el tráfico. Para mayor comodidad he optado por utilizar Windows 10, donde utilizaremos tanto Matlab y Dev-C++ para escribir sus respectivos programas como Wireshark para visualizar los archivos .pcap que contienen los flujos de tráfico.

capturados, para realizar los filtrados y la conversión del resultado final del filtrado a un archivo .txt.

## 4.4 Instalación programas

En este apartado se explicará como he descargado e instalado los programas dependiendo del sistema operativo. En el sistema operativo Linux (Ubuntu) se ha instalado el Wireshark, Nttcp e Iperf mientras que en la partición de Windows se encuentran el Wireshark, Matlab y Dev-Cpp.

### 4.4.1 Instalación programas en Ubuntu

El primer paso antes de comenzar la captura es simplemente descargarse el programa nttcp en su versión más reciente tanto en el ordenador que se encargará de transmitir como en el que se encargará de recibir. Para ello se abrirá una ventana de comando y se introducirá primero:

*“sudo apt-get update”*

Esta línea de comando simplemente actualiza la lista de paquetes disponibles y sus versiones, pero no instala o actualiza ningún paquete. Esta línea simplemente se realiza por seguridad y para evitar problemas futuros ya que se acaba de instalar Ubuntu y puede que la lista de paquetes no esté actualizada debido a esto. Cabe recordar que siempre que se esté utilizando la ventana de comandos de Ubuntu siempre antes de cada línea de comando se escribirá la palabra “sudo”, que es un programa de Linux el cual nos permitirá ejecutar esa línea de comando como un superusuario, es decir, como si fueras root, y así no salte ningún error relacionado con que no se tienen unos determinados privilegios de seguridad para realizar una acción. Cabe recordar que estas líneas de comando se deben de ejecutar teniendo una conexión a internet. Una vez ejecutado la línea de comando anterior sin ningún problema, se procederá a ejecutar la siguiente línea de comando:

*“sudo apt-get install nttcp”*

Al ejecutar esta línea de comando se instalará finalmente en nuestro ordenador el programa nttcp para realizar las medidas.

Para la instalación de la herramienta de medida Iperf tendrá un procedimiento similar a la instalación de Nttcp, ejecutando la siguiente línea de comando:

*“sudo apt-get install iperf”*

## 4.4.2 Instalación programas en Windows

Para la instalación de los programas indicados en Windows simplemente hay que ir a la página oficial del programa, descargarlo e instalarlo.

## 4.5 Características internas adaptador USB Wi-Fi TL-WN821N

Para ver la información de los dispositivos conectados a un puerto USB, como puede ser nuestro adaptador USB Wi-Fi TL-WN821N, se abrirá una ventana de comandos e introducirá la siguiente línea de comando: “lsusb”.

```
rga91@Rodrigo:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 006: ID 0cf3:7015 Atheros Communications, Inc. TP-Link TL-WN821N v3 / TL-WN822N v2 802.11n [Atheros AR7010+AR9287]
Bus 001 Device 004: ID 062a:4106 Creative Labs
Bus 001 Device 003: ID 0e8f:0021 GreenAsia Inc. Multimedia Keyboard Controller
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

*Figura 4.4. Respuesta al comando lsusb*

La ventana de comandos devolverá información sobre los dispositivos USB conectados al ordenador, correspondiendo la línea 4 a la información relacionada con nuestro adaptador USB Wi-Fi, como su creador, Atheros Communications, el modelo del dispositivo empleado y los SOC (system on a chip) mencionados en el apartado 2.1.4.

Ahora se utilizará otro comando para conocer la información del driver de Atheros. Para ello se escribirá el siguiente comando en la línea de comandos:

*“dmesg | grep ‘ath’”*

```
rga91@Rodrigo:~$ dmesg | grep 'ath'
[ 57.448675] usb 1-4: ath9k_htc: Firmware ath9k_htc/htc_7010-1.4.0.fw requested
[ 57.448722] usbcore: registered new interface driver ath9k_htc
[ 57.548912] usb 1-4: ath9k_htc: Transferred FW: ath9k_htc/htc_7010-1.4.0.fw, size: 72812
[ 57.618895] ath9k_htc 1-4:1.0: ath9k_htc: HTC initialized with 45 credits
[ 57.840380] ath9k_htc 1-4:1.0: ath9k_htc: FW Version: 1.4
[ 57.840381] ath9k_htc 1-4:1.0: FW RMW support: On
[ 57.840382] ath: EEPROM regdomain: 0x809c
[ 57.840382] ath: EEPROM indicates we should expect a country code
[ 57.840383] ath: doing EEPROM country->regdmn map search
[ 57.840383] ath: country maps to regdmn code: 0x52
[ 57.840384] ath: Country alpha2 being used: CN
[ 57.840384] ath: Regpair used: 0x52
[ 57.847879] ath9k_htc 1-4:1.0 wlx647002216fe2: renamed from wlan0
```

*Figura 4.5. Respuesta al comando “dmesg | grep ‘ath’”*

El resultado que ofrece la ventana de comandos muestra que el driver que se emplea el driver ath9k\_htc, que es un driver de código abierto, y su versión. Cabe destacar que en la última línea aparece como el driver ath9k\_htc renombra el nombre del interfaz Wi-Fi del ordenador “wlan 0” a “wlx647002216fe2”.

Por último, hay que recordar que para emplear estos dos comandos obviamente antes hay que conectar el adaptador USB Wi-Fi TL-WN821N al ordenador.

## Capítulo 5. Realización de las medidas

---

El objetivo de este apartado será explicar cómo se produce la realización de las medidas, que instrucciones utilizo en la ventana de comandos, explicación de los programas que intervienen y sus funciones y objetivos dentro del proceso. Principalmente usare el programa “nttcp” para medir la velocidad y otras características que se explicarán más adelante y realizar la transferencia de datos con datagramas UDP.

Como se explicó en la introducción, se realizarán 20 medidas en el modo infraestructura, 5 medidas en el escenario 1 por la mañana, 5 medidas en el escenario 2 por la mañana, 5 medidas en el escenario 1 por la tarde y 5 medidas en el escenario 2 por la tarde. El escenario 1 consiste en el transmisor situado al lado del router por lo que la transmisión de paquetes desde el transmisor al router será muy buena con pocas retransmisiones y a una alta velocidad, mientras que en el escenario 2 el transmisor se situará al lado del receptor alejado del router provocándose un incremento de las retransmisiones y descenso del bit rate.

En las medidas realizadas se utilizará el ordenador portátil como transmisor, ya que debido a su movilidad nos permitirá capturar el tráfico en ambos escenarios de ambos flujos existentes en la transmisión de paquetes, el flujo del transmisor al router, y el flujo del router al receptor. El ordenador fijo se convertirá en el receptor.

Para la red Wi-Fi en modo Ad-hoc se realizarán 20 medidas también, 5 para cada escenario: router y transmisor juntos por la mañana (escenario 1), router y transmisor lejos por la tarde (escenario 2), router y transmisor juntos por la tarde (escenario 3) y router y transmisor lejos por la tarde (escenario 4). En este caso habrá un único flujo que transmitirá el ordenador portátil y recibirá el ordenador fijo donde variará la posición del portátil según el escenario y el horario.

Una vez definidos los objetivos, número de medidas a realizar y escenarios posibles, se procederá a realizar las pruebas para extraer toda la información que sea útil.

### 5.1 Creación red Wi-Fi en modo Ad-hoc

Este paso solamente estará presente para la realización de las medidas en una red Wi-Fi en modo Ad-hoc. Este apartado explicará cómo se realizará la creación de una red Wi-Fi Ad-hoc empleando los adaptadores Wi-Fi USB.

Para ello se abrirá una ventana de comandos y ejecutará el comando “nm-applet”, que ejecutará el programa Network Manager que viene de serie y se activará un icono en la barra superior. Haz click en él y aparecerá una serie de opciones, entre las que estará “Editar conexión”. Haz click y aparecerá una serie de redes Wi-Fi a las que se ha conectado el ordenador anteriormente. Para añadir una nueva red Wi-Fi haz click en el signo + y aparecerá una serie de opciones para crear la red Wi-Fi que se desee:

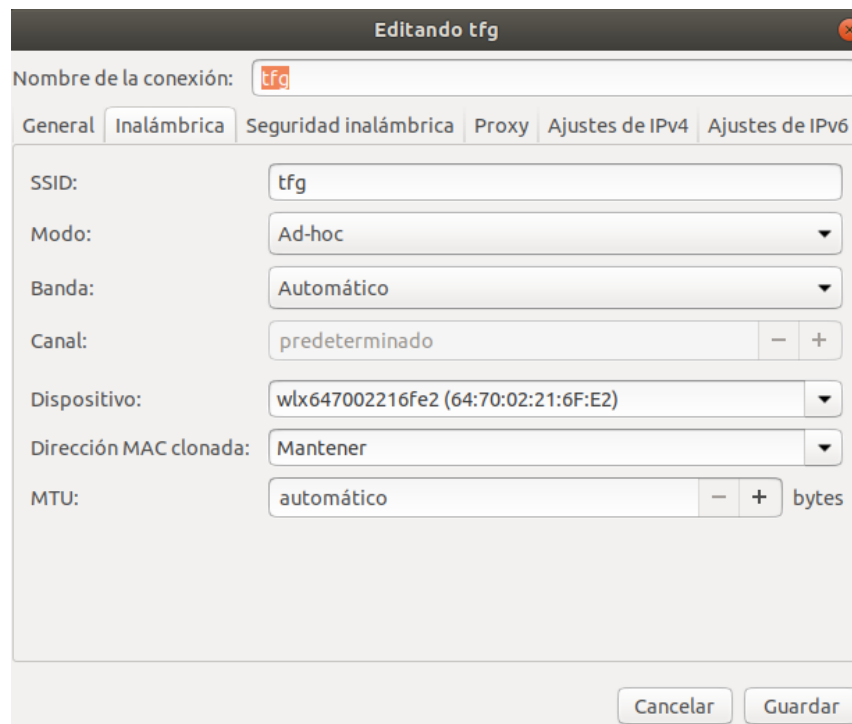


Figura 5.1. Cuadro de opciones para la creación de una red Wi-Fi

En mi caso llamaré a la red Ad-hoc “tfg” (SSID). En modo selecciona “Ad-hoc”, en Banda “Automático” aunque solo se trabajará en la banda de los 2,4 GHz. En dispositivo se elige al adaptador v USB TL-WN821N que creará la celda Ad-hoc. El resto de las opciones irán por defecto como la MTU y la dirección MAC clonada. Una vez hecho esto se guarda la configuración y se comprueba si se ha guardado correctamente comprobando la lista de redes mencionadas anteriormente:

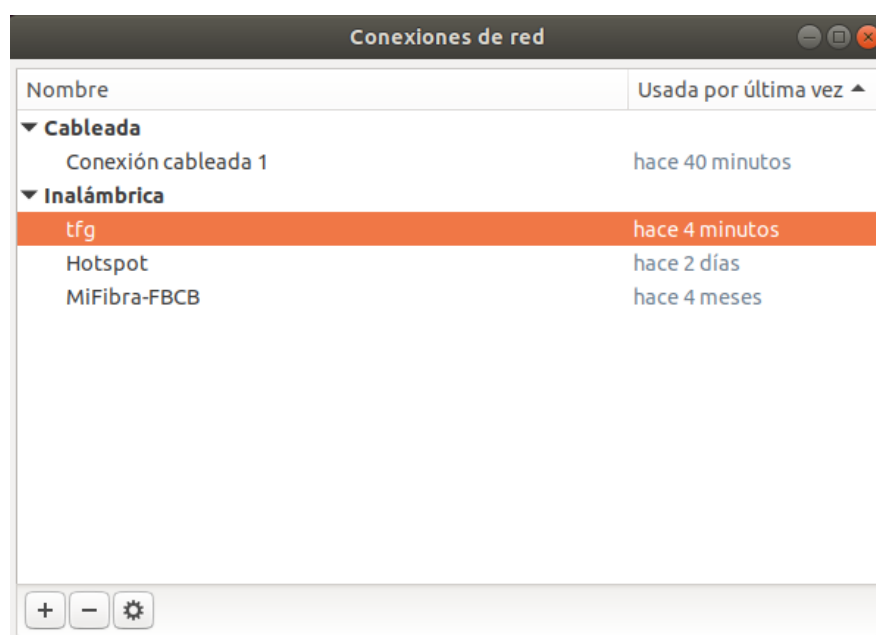


Figura 5.2. Lista de redes Wi-Fi registradas en el ordenador

Una vez hecho esto, haz click de nuevo en el icono del Network Manager y esta vez haz click en la opción “Conectarse a una red Wi-Fi oculta” y una vez ahí aparecerá una ventana en la que se seleccionará la red Wi-Fi Ad-hoc creada anteriormente y el adaptador Wi-Fi USB creará la celda Ad-hoc (No se empleará ningún tipo de codificación en este caso). Para comprobar si se ha creado correctamente la red Wi-Fi Ad-hoc se abrirá una nueva ventana de comandos y se comprobará con el comando “iwconfig” que efectivamente se ha creado:

```
rga91@Rodrigo:~$ iwconfig
lo                no wireless extensions.

enp0s31f6         no wireless extensions.

wlx64700221843a   IEEE 802.11  Mode:Monitor  Frequency:2.412 GHz  Tx-Power=20 dBm

                    Retry short limit:7   RTS thr:off   Fragment thr:off
                    Power Management:off

wlx647002216fe2   IEEE 802.11  ESSID:"tfg"
                    Mode:Ad-Hoc  Frequency:2.412 GHz  Cell: 76:3D:2E:BD:68:8B
                    Tx-Power=20 dBm
                    Retry short limit:7   RTS thr:off   Fragment thr:off
                    Power Management:off
```

*Figura 5.3. Comprobación de la creación de la red Wi-Fi Ad-hoc*

Para que el otro ordenador se una a la red Wi-Fi Ad-hoc simplemente se conectará la red Wi-Fi “tfg” como si fuera una red Wi-Fi normal.

## 5.2 Captura datos

Con los ordenadores completamente preparados y todos los programas necesarios instalados, ya se podrá comenzar a realizar las pruebas. Se encenderán ambos ordenadores receptor y transmisor y se entrará en la partición que se ha realizado previamente de Linux. Se desconectará los ordenadores de cualquier red Ethernet o Wi-Fi a la que puedan estar conectados.

### 5.2.1 Modo monitor

El primer paso que realizar será utilizar un adaptador Wi-Fi USB TL-WN821N y conectarlo en el ordenador portátil. Este adaptador USB actuará a modo de sniffer, por lo que se tendrá que inducirle en modo monitor, también conocido en castellano como modo promiscuo. El adaptador Wi-Fi USB en modo monitor permitirá escuchar todo el tráfico Wi-Fi que pueda de cualquier red Wi-Fi. Para ello, se tendrá que abrir una terminal e introducir una serie de líneas de comando:

Primero se ejecuta el comando “iwconfig” para ver el nombre de la interfaz de red que se ha asignado al adaptador Wi-Fi USB, que normalmente el nombre de interfaz de una red Wi-Fi sigue la estructura “Wlan” + un número. Una vez se sepa ya el nombre de la interfaz red, se deshabilitará con el siguiente comando:

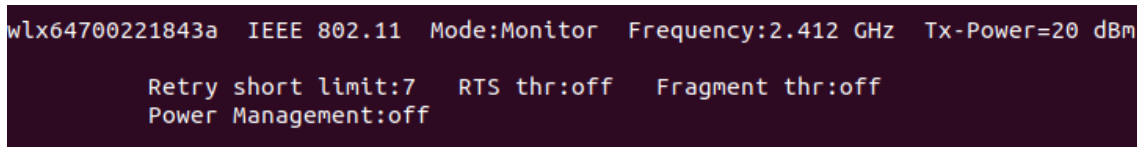
*“sudo ifconfig wlanX down”*

Acto seguido se configurará la interfaz en modo monitor:

*“sudo iwconfig wlanX mode monitor”*

Acto seguido se activará el interfaz de red y ya se tendrá la tarjeta en modo monitor:

*“sudo ifconfig wlanX up”*



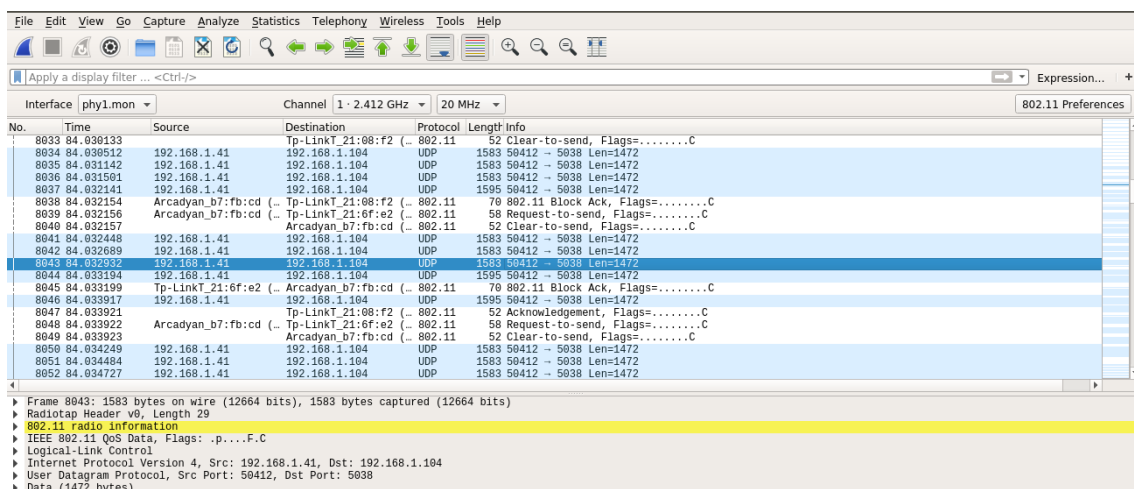
*Figura 5.4. SSID de la red Wi-Fi y algunas características*

## 5.2.2 Wireshark

Una vez se tenga la tarjeta en modo monitor, se abrirá un nuevo terminal y abrirá el analizador de protocolos Wireshark con el comando “sudo wireshark”, siempre utilizando el comando sudo para tener privilegios de superusuario.

Para capturar tráfico, simplemente se tiene que seleccionar la interfaz de red anterior la cual se encuentra en modo monitor. Una vez seleccionada, ya podrás ver como se comienza a capturar tráfico.

Un problema que puede haber es que se esté capturando tráfico en un canal que no corresponda, por lo que se tendrá que cambiar de canal. Para ello se ejecutará la siguiente línea de comando “nmcli dev Wi-Fi” en una terminal y responderá con una lista con todos los puntos de acceso Wi-Fi conocidos por el Network Manager y dará parámetros de cada uno de ellos, siendo uno el número de canal de frecuencia en el que se encuentre esa red Wi-Fi. Para elegir en que canal Wi-Fi se desea capturar tráfico, primero se habilitará la Wi-Fi Toolbar, en View -> Wireless Toolbar y aparecerá una nueva barra de herramientas donde podrás elegir el canal.



*Figura 5.5. Interfaz del Wireshark con la Wireless Toolbar*

Por último, solo queda introducir la contraseña de tu Wi-Fi en Wireshark para descryptar los paquetes que se capturen. Para ello, en la Wi-Fi Toolbar activada anteriormente se hará click “802.11 Preferences”. Se nos abrirá una ventana y se hará click en “Edit” al lado de “Decryption Keys”. Ahí se nos abrirá una ventana en la que se podrá introducir la clave de nuestro Wi-Fi. Wireshark acepta tres tipos de claves:

- Wep = la clave es analizada como una clave Wep (Ejemplo = a1:b2:c3:d4:e5).
- Wpa-pwd = la contraseña y el SSID (nombre de la red, identificador) de la red Wi-Fi se utilizan para crear una clave precompartida (Ejemplo = MyPassword:MySSID).
- Wpa-psk = la clave es analizada directamente como una clave precompartida (Ejemplo = 0102030405060708091011...6061626364)

**Importante:** Si tu red Wi-Fi usa seguridad WPA o WPA2, como era mi caso, deberás empezar a capturar paquetes con Wireshark antes de que el ordenador transmisor y el ordenador receptor se conecten a la red. Esto se debe a que WPA y WPA2 usan claves derivadas del handshake EAPOL que se producen entre el router y el ordenador al intentar este último conectarse a la red Wi-Fi. Si no se capturan todos los 4 paquetes EAPOL de esa sesión, la información quedará codificada.

### 5.2.3 Medidas con Nttcp

Con Wireshark en pleno funcionamiento, ya se podrá conectar un adaptador Wi-Fi USB TL-WN821N en el ordenador receptor y otro en el ordenador transmisor (este adaptador más el otro adaptador Wi-Fi USB que se encuentra en modo monitor haría un total de dos adaptadores conectados al portátil) y se conectarán a la red Wi-Fi. Ahora se procederá a realizar la prueba con la herramienta de medida Nttcp.

#### ➤ Receptor

En el ordenador receptor se abrirá un terminal y se escribirá la siguiente línea de comando de la cual se explicarán sus parámetros, para que el ordenador receptor esté listo para recibir la transmisión de datos:

```
nttcp -r -u -i
```

-r = indica que este ordenador es el receptor de datos

-u = se recibirá tráfico UDP

-i = este parámetro indica que, una vez recibida una transmisión de datos, el proceso no se corte y que el ordenador receptor siga en el modo de esperar más transmisiones del ordenador transmisor



### ➤ Transmisor

En el ordenador transmisor, una vez el ordenador receptor esté listo para recibir la transmisión de paquetes, se abrirá una terminal y se introducirá la siguiente línea de comando y se explicarán que significan todos los parámetros que contiene:

```
nttcp -t -u -l 1472 -n 10000 -T "dirección IP destino"
```

-t = indica que este ordenador es el transmisor de datos

-u = se transmitirá tráfico UDP

-l "Número" = indica la longitud del buffer. Para hacernos una idea, esto indica el tamaño de los datos enviados en Bytes sin contar las cabeceras.

-n "Número" = número de buffers. Esto indicaría el número de paquetes que se enviarían.

-T = escribe una línea de título en los resultados

"Dirección IP destino" = Dirección IP del ordenador receptor

Una vez ejecutada esta línea de comando, se producirá la transmisión de paquetes, y una vez acabada aparecerá en el ordenador transmisor los resultados sacados de la prueba realizada.

### ➤ Resultados

```
rodrigo@rodrigo-Aspire-A315-21:~$ nttcp -t -u -l 1472 -n 10000 -T 192.168.1.104
```

	Bytes	Real s	CPU s	Real-MBit/s	CPU-MBit/s	Calls	Real-C/s	CPU-C/s
l	14720000	3.91	0.06	30.1466	2034.0623	10003	2560.77	172781.3
1	14696448	3.91	0.08	30.0606	1482.4681	9985	2552.96	125901.5

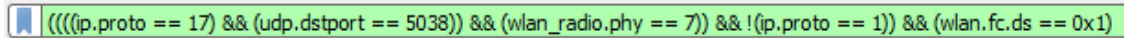
Figura 5.6. Resultados medida nttcp

La primera fila encabezada por una "l" corresponde a la información del ordenador transmisor mientras que la otra fila corresponde a la información del ordenador receptor.

La primera columna corresponde a la longitud del buffer en bytes, la segunda representa el tiempo real en segundos, la tercera el tiempo de CPU en segundos, la cuarta el bitrate real en Mbps, la quinta el bitrate de la CPU en Mbps, la sexta el número de llamadas, la séptima el ratio de llamadas por segundo real y la última el ratio de llamadas por segundo de la CPU.

## 5.2.4 Filtrado

Una vez capturada la transmisión de paquetes, se tendrán que filtrar los paquetes que nos interesen y pasarlo a un formato de fácil manejo, como puede ser .txt. Cada medida realizada se dividirá en dos archivos, uno representa el flujo de paquetes desde el transmisor al router y el otro el flujo desde el router al receptor. Para ello se utilizará la función de filtrado de Wireshark. Para ello sólo se tendrá que introducir los filtros en el buscador de la parte superior:



```
(((ip.proto == 17) && (udp.dstport == 5038)) && (wlan_radio.phy == 7)) && !(ip.proto == 1)) && (wlan.fc.ds == 0x1)
```

*Figura 5.7. Filtros empleados*

En la imagen superior se puede apreciar los filtros empleados que son:

- `ip.proto == 17` → filtra solo datagramas UDP
- `udp.dstport == 5038` → filtra tráfico cuyo puerto de destino sea 5038, el puerto que usa el programa nttcp.
- `Wlan_radio.phy == 7` → Indica que filtra tráfico 802.11.n
- `!ip.proto == 1` → Elimina los paquetes ICMP ya que por algún motivo Wireshark lo interpreta que son paquetes UDP
- `Wlan.fc.ds == 0x1` → Filtra los paquetes del flujo Transmisor → Router. Cuando se quiera obtener el archivo con los datos del flujo Router → Receptor simplemente se tendrá que cambiar el valor de filtro `wlan.fc.ds == 0x2`

Para el filtrado de tramas en una red Wi-Fi en modo Ad-hoc ya no se empleará el filtro “Wlan.fc.ds” debido a que solo hay un único de flujo de tráfico de datos.

Una vez realizado el filtrado, ya exportar los datos a formato .txt. Antes de nada, una vez se tengan representados los paquetes que se deseen, se deberá elegir que datos se van a necesitar en nuestras medidas y abrir las ventanas donde se encuentran. Por ejemplo, en nuestro caso se querrá saber la velocidad de transmisión, potencia de señal, identificador IP de los paquetes, si el paquete es una retransmisión o no y el tamaño del paquete. Para ello se tendrá que desplegar las ventanas “802.11 radio information” y “Internet Protocol Version 4”:

```

> Frame 6419: 1583 bytes on wire (12664 bits), 1583 bytes captured (12664 bits)
> Radiotap Header v0, Length 29
▼ 802.11 radio information
  PHY type: 802.11n (7)
  MCS index: 3
  Bandwidth: 20 MHz (0)
  Short GI: False
  Data rate: 26.0 Mb/s
  Channel: 1
  Frequency: 2412MHz
  TSF timestamp: 390942871
  > [Duration: 476µs]
> IEEE 802.11 QoS Data, Flags: .p.....TC
> Logical-Link Control
▼ Internet Protocol Version 4, Src: 192.168.1.41, Dst: 192.168.1.104
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 1500
  Identification: 0x4553 (17747)
  > Flags: 0x4000, Don't fragment
  Time to live: 64
  Protocol: UDP (17)
  Header checksum: 0x6bdc [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.1.41
  Destination: 192.168.1.104
> User Datagram Protocol, Src Port: 47171, Dst Port: 5038
> Data (1472 bytes)

```

*Figura 5.8. Formato de la trama en Wireshark*

Una vez hecho esto, se podrá ya pasar los datos a archivo .txt. Para ello haz click en File → Export Packet Dissections → As plain text. Se abrirá una ventana y se comprobará que este seleccionada la opción de mostrar cómo se ha elegido, con los filtros elegidos y las ventanas abiertas elegidas. En mi caso esta seleccionado la opción “Packet Summary line” que añade una línea antes de cada paquete indicando una serie de características sobre este y es la opción por defecto. Por último, se guardará el archivo en el directorio que se desee y ya estaría listo el archivo .txt de la captura.

Cabe destacar que la representación de Wireshark de las tramas obtenidas es una aproximación, al ser una trama A-MPDU todas las tramas de un grupo pertenecen a ella compartiendo la misma cabecera física. En Wireshark aparecerán como tramas independientes (aunque bien es cierto que hay un campo que te indica que forma parte de una A-MPDU) con la misma cabecera física siendo la última trama MPDU de esta ráfaga la que contenga información de la potencia de señal recibida de la A-MPDU y todas ellas con una cabecera MAC que las diferencia. (Ver apartado 2.6.2 A-MPDU)

## 5.3 Obtención datos con C

Una vez se tenga la captura en formato texto representada de la manera deseada, se deberá extraer únicamente los datos con los que se va a trabajar. Para ello se creará una serie de tablas que contengan esos datos y luego esas tablas se pasarán a Matlab para realizar la representación gráfica. Los datos que se van a obtener serán la velocidad de transmisión, potencia de señal, si una trama se va a retransmitir y si una trama es una “retransmisión fantasma”. Todo esto, la obtención de las tablas y su consiguiente explicación detallada de estas se explicará más adelante.

Primero de todo, hay que aclarar que para la obtención de datos y creación de tablas se va a utilizar un programa .C creado con Dev-Cpp. Se ha optado por el lenguaje de programación C por ser un lenguaje sencillo y con el que ya tengo conocimientos sobre él.

El nombre del programa .C es “SacaValoresFinal.c” y lo podréis encontrar en el anexo.

El programa se puede dividir en 3 partes: inicialización, lectura y obtención de datos y escritura. La parte de inicialización es básicamente donde se definen todas las variables, archivos que se van a abrir, y se inicializan algunas variables que sean necesarias. Es en esta parte donde está el código en el que se va a pedir el archivo sobre el cual se va a realizar la extracción de datos. Es, por tanto, el único mensaje que verá el usuario una vez ejecutado el programa. Cabe destacar que el programa y el archivo deben estar en el mismo directorio para que el programa encuentre el archivo designado a realizar la extracción de datos.

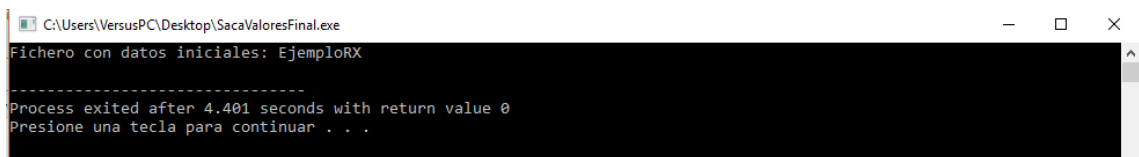


Figura 5.9. Vista del usuario al utilizar el programa

### 5.3.1 Lectura y extracción de datos

Una vez indicado el archivo a leer, procederé a explicar la manera de funcionamiento del programa y la lectura del archivo. El programa a través de un ciclo while como el que aparece en la imagen inferior, va leyendo línea a línea del archivo hasta que no haya más líneas en el archivo. La condición del while que aparece en la imagen, significa que leerá esa línea hasta que llegue a un retorno de carro. Una vez llegue a un retorno de carro, toda la línea que habrá leído la guardará en el valor tipo char “input”.

```
while(fscanf(myFile, "%[^\n]", input) != EOF)
```

Figura 5.10. Ciclo While para la lectura del archivo

El programa irá leyendo línea por línea y escogerá las líneas donde está la información que nos interesa. Pero ¿Cómo hará esto? El archivo .txt de la captura sigue una pauta. Cada paquete tiene un tamaño predeterminado. La herramienta de medida nttcp presenta cuatro tipos de tramas de distintos tamaños en Bytes. Los tamaños son de 1595, 1583, 127 y 115 Bytes. Mirando el archivo texto de la captura se observará que en la segunda línea del documento aparece información del primer paquete como por ejemplo el tamaño del paquete. Esto se

repetirá en todas las tramas contenidas que en todos los párrafos pertenecientes a cada trama la segunda línea tendrá el mismo tipo de información. Leyendo esta línea ya se sabrá cuantas líneas ocupará cada trama y se podrá obtener los datos.

No.	Time	Source	Destination	Protocol	Length	Info
3059	38.160093	192.168.1.104	192.168.1.40	UDP	115	41062 → 5038 Len=4

*Figura 5.11. Primeras dos líneas de cada trama*

Para ello se utilizará el contador “i”, un contador que contará las líneas del archivo .txt de la captura, y una serie de variables tipo integer que cuando el contador “i” tengan el mismo valor que esas variables “i = variable” indicará que esa línea se deberá analizar para la extracción de datos.

El primer caso donde se da una comparación de este estilo es con la variable “s”. La variable “s” indica prácticamente el comienzo de cada trama, apuntando a la segunda línea de información de cada trama que se ha mencionado anteriormente, donde se encuentra el tamaño de esa trama.

En la imagen inferior (Figura 5.12), se puede ver la estructura de comparación para la variable “s”. Para la primera trama, la variable “s” estará inicializada en el valor 1 (el valor 0 sería la primera línea). Una vez se cumple la condición del if, se escaneará la línea leída y se extraerá el valor del tamaño de la trama con la función sscanf, indicándole que palabra corresponde al valor deseado y el tipo de valor que corresponde, en este caso integer, asignándolo a la variable “longitud”. En este caso sscanf pedirá asignar a la variable “longitud” el sexto dato de esa línea. (Con %\*s se desecha la palabra y no se tendrá en cuenta)

```
if (i==s){
    sscanf(input,"%*s %*s %*s %*s %*s %i",&longitud);
    if(longitud == 1595 || longitud == 127 || longitud == 1579 || longitud == 111){
        ratecont = s+8;
    } else{
        retransmision = s + 13;
    }
}
```

*Figura 5.12. Bloque de código para conocer tamaño de la trama*

Una vez se sepa el tamaño de la trama, se comprobará que tipo de trama es, ya que no todas las tramas pueden aportar los parámetros que se deseen. En este caso se realizará un if para determinar los tamaños, si su tamaño es 127, 111, 1579 o 1595 se indicará el valor de la variable “ratecont” con respecto al valor de “s” o “i”, relacionado con la extracción de la velocidad de transmisión. Si no cumple con esos tamaños, se irá directamente a comprobar si es una retransmisión.

Antes de nada, hay que aclarar que estos “saltos” para ver donde hay que analizar se realizan como podréis comprobar teniendo en cuenta valores anteriores por lo que un error en el código o en el formato del archivo .txt podría llevar a un error total en las soluciones.

Las tramas que contienen más información sobre el estado del canal son las de tamaño 127, 111, 1579 y 1595 Bytes, ya que tienen lo mismo que las demás tramas más otros parámetros, por lo que seguiré los saltos realizados en caso de que sea una trama de 127, 111, 1579 y 1595 Bytes.

El porqué de estas diferentes longitudes para las tramas depende de la misma herramienta de medidas Nttcp y el tipo de red Wi-Fi utilizado (Ad-hoc e infraestructura). Primero, Nttcp marca el inicio y finalización de datos de una transmisión de datos con una trama de pequeña longitud de 115 Bytes (127 Bytes si contiene información sobre la potencia de señal recibida) y las demás tramas con una longitud de 1583 Bytes (1595 Bytes si contiene información sobre la potencia de señal recibida) que tendrán un campo de datos de 1472 Bytes y se enviarán 10000 paquetes de este estilo como se definió a la hora de ejecutar Nttcp en la ventana de comandos. Por último, el motivo por el que hay tramas con menor tamaño es debido a que en la red Wi-Fi en modo Ad-hoc no se emplea ningún tipo de codificación siendo la cabecera física más ligera, 16 Bytes más ligero.

Continuando, el siguiente salto caerá en la línea donde se define la velocidad de transmisión de la trama. Cabe destacar que todas las tramas tienen definido el parámetro de velocidad de transmisión, pero debido a que todas las tramas de una misma ráfaga de tramas tienen la misma velocidad de transmisión, con que se coja la velocidad de transmisión de la trama de tamaño 127 o 1595 Bytes, que corresponde a la última trama de la ráfaga, ya se podría obtener el dato deseado.

```
if (i == ratecont){  
    Antena = ratecont +3;  
    sscanf(input,"%*s %*s %f",&rate[j]);  
}
```

*Figura 5.13. Bloque de código para conocer la velocidad de transmisión de la A-MPDU*

En la imagen superior, se puede ver el if que una vez la lectura de línea (contador “i”) llega al valor de la variable “ratecont”, se encarga de indicar el siguiente salto, en este caso la línea donde se encuentra definida la potencia de la señal de la ráfaga, solo presente en tramas de tamaño de 127, 111, 1579 y 1595 Bytes, con la variable “Antena” y de analizar la línea y extraer la velocidad de transmisión de la ráfaga con un sscanf. La velocidad de transmisión se guardará en el array “rate[]” y será una vez completada la lectura total del archivo, el array que contendrá todas las velocidades de transmisión de las ráfagas, por lo que será unos de los datos que se representarán gráficamente en Matlab.

El siguiente salto solo estará presente en las tramas de tamaño de 127 o 1595 Bytes y es donde se extraerá la potencia de la señal recibida de una ráfaga de tramas A-MPDU. En la imagen inferior, se pondrá una condición “if” similar a la anterior. Como podréis una vez se cumpla la condición indicará el siguiente salto, en el cual se comprobará si esa trama en concreto es una retransmisión o no, y se analizará la línea leída y extraerá el valor de la potencia de señal

recibida con sscanf. Este valor al igual que el de velocidad de transmisión, será uno de los datos que se necesitarán más adelante para mostrar la información del canal Wi-Fi, por lo que se guardará en el array “pot[]”. Los arrays “pot[]” y “rate[]” tendrán que ser del mismo tamaño (mismo número de tramas de tamaño 127, 111, 1579 y 1595 Bytes), y esta asignación a cada posición del array se realizara con el contador “j” que incrementará en 1 su valor cada vez que una trama de 127, 111, 1579 o 1595 bytes sea leída.

```

if (i == Antena){
    sscanf(input,"%*s %*s %*s %i",&pot[j]);
    retransmision= Antena + 3;
    j = j+1;
}

```

*Figura 5.14. Bloque de código para conocer la potencia de señal recibida de la A-MPDU*

El siguiente salto estará presente en todas las tramas e indicará si esa trama es una retransmisión o no. La estructura “if” de este caso aparecerá representada en la imagen siguiente:

```

if(i == retransmision){
    sscanf (input,"%*s %*s %*s %*s %*s %s",Rtxstr);
    flujo = flujo +1;
    if(strcmp(Rtxstr,NoRtx) == 0 || strcmp(Rtxstr,NoRtx2) == 0 || strcmp(Rtxstr,NoRtx3) == 0){
        rtxindicador = 1;
        rtx=rtx +1;
    } else {
        rtxindicador = 0;
    }
    if (longitud == 1595 || longitud == 127 || longitud == 1579 || longitud == 111){
        rtxflujo[j-1][0]=rtx;
        rtxflujo[j-1][1]=flujo;
        flujo=0;
        rtx=0;
    }
    id = retransmision +7;
}

```

*Figura 5.15. Bloque de código para saber si la trama analizada es una retransmisión*

Como se puede sigue la misma estructura como todos los casos anteriores: se alcanza la línea deseada. Se analiza y se extraen los datos deseados. En esta parte se detallará cuantas tramas de una ráfaga de tramas son retransmisiones. Para ello primero se hará una comparación con el dato extraído, si el dato extraído es una retransmisión, se incrementará un contador de retransmisiones “rtx” que contará el número de retransmisiones en una misma ráfaga. El

número total de tramas quedará descrito en el valor de la variable “flujo”, que actúa como un contador.

**IEEE 802.11 QoS Data, Flags: .p..R..TC**  
*Figura 5.16. Cabecera IEEE 802.11 de una retransmisión*

El final de una ráfaga de tramas viene determinado por la lectura de una trama de tamaño de 127, 111, 1579 y 1595 Bytes. Una vez recibida una trama de este tipo, los valores en ese momento en “flujo” (número de tramas de la ráfaga) y de “rtx” (número retransmisiones de esa ráfaga) se guardarán en la matriz de dos columnas “rtxflujo”, correspondiendo la primera columna a la variable “rtx” y la segunda columna a la variable “flujo”. Una vez guardados los valores, se reiniciarán los contadores de “flujo” y “rtx” a valor 0. Una vez pasadas todas las condiciones “if”, se indicará el siguiente salto “id”. La variable “rtxindicador” indicará si esa trama es una retransmisión y será utilizada más adelante.

El próximo salto estará presente en todas las tramas y será el último. El objetivo de este salto es principalmente obtener el identificador IP de la trama, aunque se realizarán más funciones como se explicará más adelante. Como siempre se llegará a la condición “if”, se leerá la línea del archivo y después se analizará y se extraerá en este caso el identificador. El identificador es una variable tipo char y se guardará en el array “listaid[]” que contendrá todos los identificadores en formato char. Ahora la condición “if == id” se dividirá en cuatro partes, cada una destinada a realizar un trabajo.



```

if(i == id){
    sscanf(input,"%*s %*s %s",listaid[conttrama]);

    if (conttrama == 0){
        relacion[0][0]= identificador;
        identificador = identificador +1;
    }else{
        for(k=0;k<conttrama;k++){
            if(strcmp(listaid[k], listaid[conttrama]) == 0 && relacion[k][2] == 0){
                repe=1;
                original=k;
                break;
            }
        }

        if (repe==0){
            relacion[conttrama][0]=identificador;
            identificador = identificador +1;
        }else if(repe==1){
            relacion[original][2]=1;
            relacion[conttrama][0]=relacion[original][0];
            repe=0;
        }
    }
}

```

*Figura 5.17. Bloque de código para crear una lista de todas las tramas ordenadas por orden aparición e indicando el número de veces que se retransmite*

El primer “if” simplemente servirá para indicar si es la primera trama o no. En el caso de que así lo sea, se guardará la variable “identificador” de valor 1 en la primera columna, primera posición, de la matriz “relacion[][]” y se incrementará la variable “identificador” en 1. Esta variable “identificador” vendría a realizar la misma función que los valores de “listaid[]” pero en formato integer gracias a lo cual es más fácil trabajar. En el caso de que no sea la primera trama, se entrará en un ciclo “for” junto a un “if” que irá comparando el identificador nuevo obtenido de la trama, con la lista de identificadores obtenidos anteriores. Si se da el caso de que efectivamente esa trama se repite, es decir, esta trama se perderá y se retransmitirá y no ha sido notificada en la tabla de relaciones ya que puede haber más de una retransmisión de una misma trama por lo que la columna de la matriz “relación[][2]”, nos indicará si ya había sido indicada anteriormente que esa trama en particular se retransmitirá, por lo que habría que buscar más adelante en la lista una trama con el mismo identificador, es decir una retransmisión de la trama original que también se perderá y se volverá a retransmitir pero aún no se había marcado como retransmisión. Los datos que representen las tablas que se ofrezcan como resultados del programa .C como “relacion[][]” se explicarán más adelante.

Una vez, comprobado que se repite y marcada la trama de la cual es una retransmisión, se asignará a la variable “repe” a valor 1 indicando que es una retransmisión y se parará el ciclo “for”, guardando en la variable “original” la posición de la trama de la cual se produce la pérdida y posterior retransmisión.

Una vez fuera del ciclo “for”, se entrará en una estructura condicional tipo “if” en la cual se comprobará con la variable “repe” si es una retransmisión o no. En el caso de que no sea una retransmisión, entonces corresponderá a una trama nueva transmitida por primera vez por lo que se le asignará un identificador nuevo único (asignado en la primera columna de la matriz relación), incrementando después el valor de la variable “identificador” en 1. La variable “conttrama” es un contador que contará el número de tramas del archivo .txt de la captura y se utilizará para determinar la posición de las tramas y que haya un orden. Para resumir, la matriz relacion[][] tendrá el número de filas será igual al valor final de la variable “conttrama”.

En el caso de qué sí sea una retransmisión, en la matriz relacion[], concretamente en su tercera columna (relacion[][2]), en la posición de fila que indica la trama la cual se va a retransmitir más adelante se marcar con un 1, y en la trama actual que es la retransmisión de la trama anterior se le asignará el mismo identificador de la trama anterior. Por último, se reinicializará la variable “repe” a valor 0.

Acto seguido, una vez registrado el identificador de la trama e identificar si es la retransmisión de una trama, para la tabla de resultados además de estos valores se van a indicar la velocidad de transmisión de cada trama y su potencia de señal. Como todas las ráfagas van a la misma velocidad de transmisión y tienen la misma potencia de señal, solamente se tendrá que copiar un valor en toda la ráfaga, valor que ya se había obtenido anteriormente. Para ello se realizará una condición tipo “if” (imagen inferior) que permita el acceso a un ciclo “for” solo cuando la trama leída sea de tamaño de 111, 127, 1579 o 1595 Bytes (que son las tramas delimitan el final de una ráfaga). El ciclo “for” se encargará de escribir en la matriz relacion[] la potencia de señal y en otro array, raterelacion[], la velocidad de transmisión. La velocidad de transmisión va aparte debido a que es un valor tipo “float” y no “integer” como los demás variables de la matriz relacion[].

```
if(longitud == 1595 || longitud == 127 || longitud == 1579 || longitud == 111){
    for(k=indice;k<=conttrama;k++){
        relacion[k][1]=pot[j-1];
        raterelacion[k]=rate[j-1];
    }
    indice = conttrama +1;
}
```

*Figura 5.18. Bloque de código para asignar la velocidad de transmisión y potencia de señal a cada trama en la lista de tramas mencionada anteriormente*

La variable “índice” indica el inicio de cada ráfaga y que coincide con el final de la anterior ráfaga + 1. El ciclo “for” asignará valores desde el valor marcado por “índice” hasta el número de trama actual.

El siguiente conjunto de código servirá para determinar si esta trama es una “retransmisión fantasma” o no. Como inciso, una retransmisión fantasma consiste en una trama

que es una retransmisión, pero la trama de la cual es una retransmisión no aparece en la captura del flujo de datos ya sea por cualquier motivo.

Para ello la trama actual como se ha indicado deberá ser una retransmisión y se indicará con la variable “rtxindicador” mencionada anteriormente. Esto se realizará con una condición tipo “if”. Básicamente lo que viene después es un ciclo “for” que junto a una condición tipo “if” se compara la trama retransmitida actual con todas las tramas recibidas hasta ese momento. Si aparece una trama con el mismo identificador que el de la trama actual, se actualizará el valor de la variable “contcomp” a uno distinto de 0. Si una vez acabado el ciclo “for” el valor de la variable “contcomp” es 0, se tratará de una “retransmisión fantasma” y si es al revés, es decir una retransmisión normal, “contcomp” tendrá cualquier otro valor. Al inicio de este bloque de código, se asignará el valor de 0 a la variable “contcomp”.

```
contcomp=0;
if(rtxindicador == 1){
    for(k=0;k<conttrama;k++){
        if(relacion[k][0] == relacion[conttrama][0]){
            contcomp= contcomp +1;
        }
    }
    if(contcomp == 0){
        relacion[conttrama][3] = 1;
    }
}
```

Figura 5.19. Bloque de código para indicar en la lista de tramas si una trama es una “retransmisión fantasma”

Por último, una vez extraídos todos los datos necesarios de una trama, se tendrá que producir el salto a la siguiente trama, realizado con un “if”. Las tramas de tamaño 1595, 1583, 1567 y 1579 Bytes tendrán el mismo valor de salto, mientras que las tramas de tamaño de 111, 99, 127 y 115 Bytes será mucho menor. Esto es debido al tamaño del campo de datos.

```
if (longitud == 1595 || longitud == 1583 || longitud == 1579 || longitud == 1567){
    s = id + 103;
} else {
    s = id + 12;
}
```

Figura 5.20. Bloque de código para indicar el comienzo de la siguiente trama

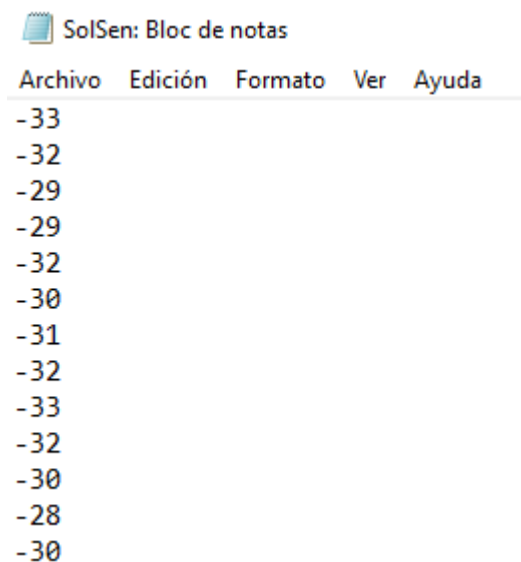
Una vez finalizado el ciclo “While”, el archivo .txt de la captura se habrá leído por completo habiéndose extraído los datos deseados y se procederá a escribir esos datos en una serie de archivos de texto.

### 5.3.2 Escritura de resultados

El primero de ellos en escribirse será el archivo de texto “SolSen.txt”. Este archivo de texto contiene una serie de números que corresponde a la potencia señal recibida en el orden de llegada de las ráfagas de tramas. Se utilizará un ciclo “for” para escribir los datos en todos los casos. El valor de la potencia de señal recibida se halla en el array “pot[]”.

```
myFile2 = fopen("SolSen.txt", "w");  
for(i=0;i<j;i++){  
    fprintf(myFile2,"%i\n",pot[i]);  
}  
fclose(myFile2);
```

Figura 5.21. Bloque de código para escribir el archivo “SolSen.txt”



SolSen: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
-33				
-32				
-29				
-29				
-32				
-30				
-31				
-32				
-33				
-32				
-30				
-28				
-30				

Figura 5.22. Visualización del archivo SolSen.txt

El siguiente archivo de texto que se escribirá será “SolRate.txt”. Este archivo de texto contendrá las velocidades de transmisión de las ráfagas de tramas que se han capturado en orden de llegada. La velocidad de transmisión de cada ráfaga de tramas está guardado en el array “rate[]”.

```

myFile3 = fopen("SolRate.txt", "w");

for(i=0;i<j;i++){

    fprintf(myFile3,"%f\n",rate[i]);

}

fclose(myFile3);

```

Figura 5.23. Bloque de código para escribir el archivo "SolRate.txt"

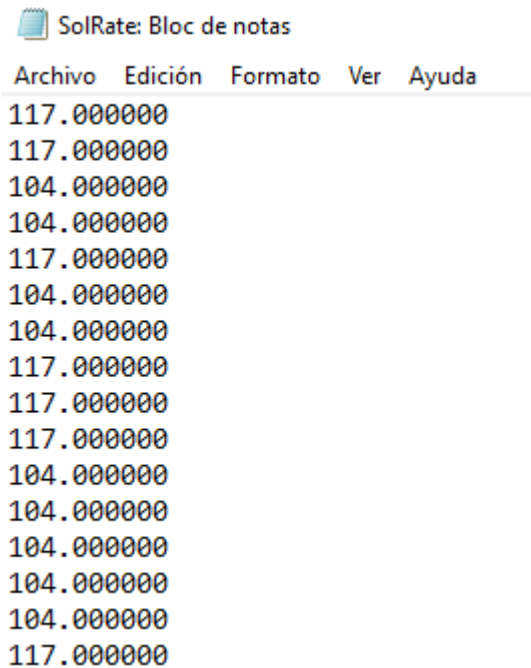


Figura 5.24. Visualización del archivo SolRate.txt

El tercer archivo de texto que se escribirá es "SolRtxflujo.txt". En este archivo de texto se representará el número de tramas de una ráfaga que son retransmisiones frente al número total de tramas de una ráfaga, todo ello ordenado según el orden de captura de las ráfagas. Se reflejará en dos columnas, la primera representando el número de tramas que son una retransmisión y en la segunda columna el número total de tramas de esa ráfaga. Cabe destacar, que estos datos no se usarán en Matlab más adelante. El objetivo de este archivo es comprobar si hay errores como por ejemplo un mal filtrado. Los datos para escribir este archivo de texto se extraerán de la matriz "rtxflujo[][]", donde la primera columna representa las retransmisiones y la segunda columna el número total de tramas de la ráfaga.

```

myFile4 = fopen("SolRtxflujo.txt", "w");

for(i=0;i<j;i++){

    fprintf(myFile4,"%i %i\n",rtxflujo[i][0], rtxflujo[i][1]);

}

fclose(myFile4);

```

Figura 5.25. Bloque de código para escribir el archivo "SolRtxflujo.txt"

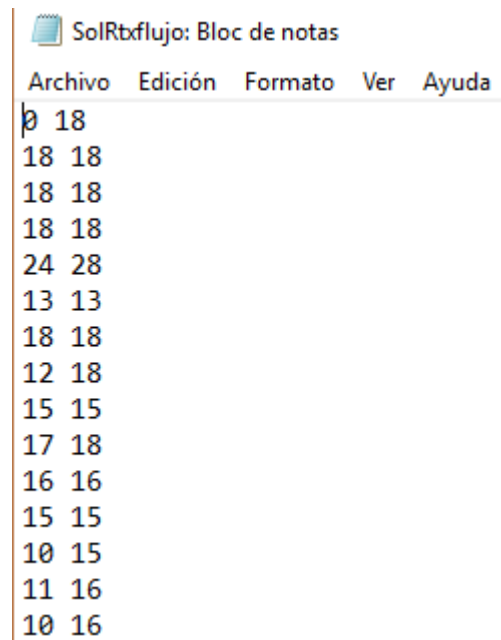


Figura 5.26. Visualización del archivo SolRtxflujo.txt

El último archivo de texto que se escribirá es “Relaciones.txt”. Este archivo de texto contiene información de cada trama de la captura en el orden de llegada. Tiene cinco columnas, siendo primera el identificador de trama obtenido de la primera columna de la matriz “relación[][]”, la segunda la potencia de señal recibida obtenida de la segunda columna de la matriz “relación[][]”, la tercera indica si la trama se va a retransmitir obtenido de la tercera columna de la matriz “relación[][]”, la cuarta la velocidad de transmisión obtenido del array “raterelacion[]” y la quinta indica si es una “retransmisión fantasma” obtenido de la cuarta columna de la matriz “relación[][]”.

```
myFile5 = fopen("Relaciones.txt", "w");
for(i=0;i<contramaj;i++){
    fprintf(myFile5,"%i %i %i %f %i\n",relacion[i][0], relacion[i][1], relacion[i][2], raterelacion[i], relacion[i][3]);
}
fclose(myFile5);
```

Figura 5.27. Bloque de código para escribir el archivo “Relaciones.txt”

	Archivo	Edición	Formato	Ver	Ayuda
1	-33	1	117.000000	0	
2	-33	1	117.000000	0	
3	-33	1	117.000000	0	
4	-33	1	117.000000	0	
5	-33	1	117.000000	0	
6	-33	1	117.000000	0	
7	-33	1	117.000000	0	
8	-33	1	117.000000	0	
9	-33	1	117.000000	0	
10	-33	1	117.000000	0	
11	-33	1	117.000000	0	
12	-33	1	117.000000	0	
13	-33	1	117.000000	0	
14	-33	1	117.000000	0	
15	-33	1	117.000000	0	
16	-33	1	117.000000	0	
17	-33	1	117.000000	0	
18	-33	1	117.000000	0	
1	-32	1	117.000000	0	
2	-32	1	117.000000	0	
3	-32	1	117.000000	0	
4	-32	1	117.000000	0	
5	-32	1	117.000000	0	
6	-32	1	117.000000	0	
7	-32	1	117.000000	0	

Figura 5.28. Visualización del archivo Relación.txt.

## 5.4 Matlab

Una vez extraídos los datos necesarios a través del programa .C y con los archivos de texto que los contienen, se procederá con Matlab a realizar una representación gráfica de los mismos. Para ello se utilizará el script “DibujaFinal.m”. Para ejecutarlo simplemente se tendrá que llamar en la ventana de comandos de Matlab con tres datos de entrada que corresponden con los tres archivos de texto necesarios para la representación de datos que son: “SolSen.txt”, “SolRate.txt” y “Relaciones.txt” en ese orden. Ejemplo para llamada de función en la ventana de comandos: `DibujaFinal('SolSen.txt','SolRate.txt','Relaciones.txt')`.

El primer trabajo del script será leer los valores contenidos en los ficheros y guardarlos en unas variables. Para ello se deberá especificar en el “formatSpec” el tipo de variable leída (integer o float), y en el caso de la lectura del archivo de texto “Relación.txt” habrá que definirlo para cada columna, y añadir un parámetro nuevo “sizeC” que indica el tamaño que se deberá esperar en la lectura, en este caso 5 columnas de valores y en el valor de filas infinitas para que no haya problemas (en mi caso Matlab lo leerá como cinco filas con infinitas columnas, por lo que luego traspondré la matriz para que quede reflejado como he dicho).

```

fileID = fopen(x,'r');
formatSpec = '%d';
A = fscanf(fileID,formatSpec);

```

*Figura 5.29. Lectura archivo SolSen.txt*

```

fileID2 = fopen(y,'r');
formatSpec2 = '%f';
B = fscanf(fileID2,formatSpec2);

```

*Figura 5.30. Lectura archivo SolRate.txt*

```

fileID3 = fopen(z,'r');
formatSpec3 = '%d %d %d %f %d';
sizeC = [5 Inf];
C = fscanf(fileID3,formatSpec3,sizeC);
C = C';

```

*Figura 5.31. Lectura archivo Relaciones.txt*

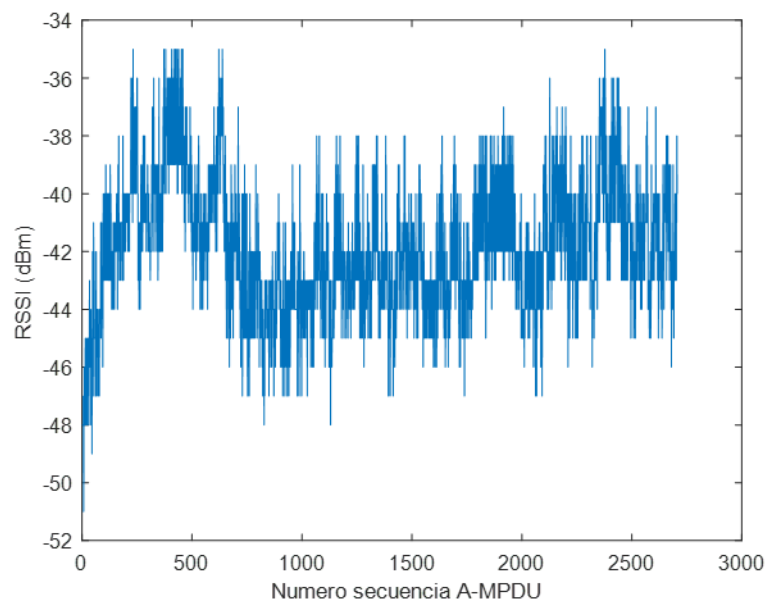
La primera gráfica que se representará contendrá la potencia de señal recibida de cada ráfaga de tramas (A-MPDU), utilizando los datos obtenidos de “SolSen.txt”. Para ello se usará el comando “plot”.

```

figure(1)
plot(A)
ylabel('RSSI (dBm)')
xlabel('Número secuencia A-MPDU')
fclose(fileID);

```

*Figura 5.32. Bloque de código para la representación de la potencia de señal de cada A-MPDU*



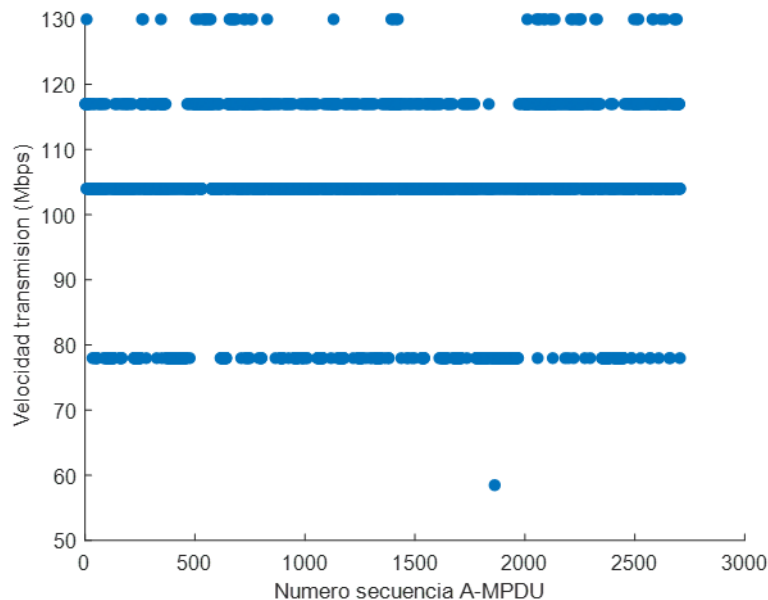
*Figura 5.33. Representación de la potencia de señal de cada A-MPDU*



La segunda gráfica que se representará contendrá la velocidad de transmisión de cada ráfaga de tramas (A-MPDU), utilizando los datos obtenidos en “SolRate.txt”. Para ello se usará el comando “scatter”.

```
Tam=size(B);
Y=1:1:Tam;
figure(2)
scatter(Y,B,'filled')
ylabel('Velocidad transmisión (Mbps)')
xlabel('Número secuencia A-MPDU')
fclose(fileID2);
```

*Figura 5.34. Bloque de código para la representación de la velocidad de transmisión de cada A-MPDU*



*Figura 5.35. Representación de la velocidad de transmisión de cada A-MPDU*

La tercera gráfica que se representará será el número total de tramas que se retransmiten a una velocidad de transmisión y el número de esas tramas que se retransmitirán. Para ello se utilizarán los datos de “Relaciones.txt”, los cuales se emplearán para obtener los valores necesarios. Los valores necesarios se detallarán en la matriz “Sol” de tres columnas, la primera columna conteniendo las velocidades de transmisión únicas, la segunda el número total de tramas que se van a transmitir a esa velocidad de transmisión y la tercera el número de total de tramas transmitidas a esa velocidad de transmisión que se van a retransmitir.

Para ello se empleará un ciclo “for” que se encargará de comparar velocidades de transmisión y cuando coincidan se incrementará el valor de número total de tramas en 1, se indicará que es una velocidad de transmisión repetida y se incrementará en 1 el valor de las tramas que se retransmitirán en caso de que se retransmita en un futuro. Todos estos valores

(velocidad de transmisión (cuarta columna) y si la trama se retransmitirá (tercera columna)) están incluidos en la matriz C que contiene todos los datos de “Relaciones.txt”.

En el caso de que no sea una velocidad de transmisión repetida, se incrementará en 1 las filas de la matriz Sol y se registrará esa velocidad de retransmisión con los datos concernientes a esa primera trama a esa velocidad de transmisión.

```

for i = 1:maximo
    for j = 1:Rateunico
        if Sol(j,1) == C(i,4)
            Sol(j,2) = Sol(j,2) + 1;
            Sol(j,3) = Sol(j,3) + C(i,3);
            repe=1;
        end
    end

    if repe==0
        Sol(Rateunico,1) = C(i,4);
        Sol(Rateunico,2) = 1;
        Sol(Rateunico,3) = C(i,3);
        Rateunico=Rateunico + 1;
    else
        repe=0;
    end
end

```

*Figura 5.36. Bloque de código para la obtención del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad*

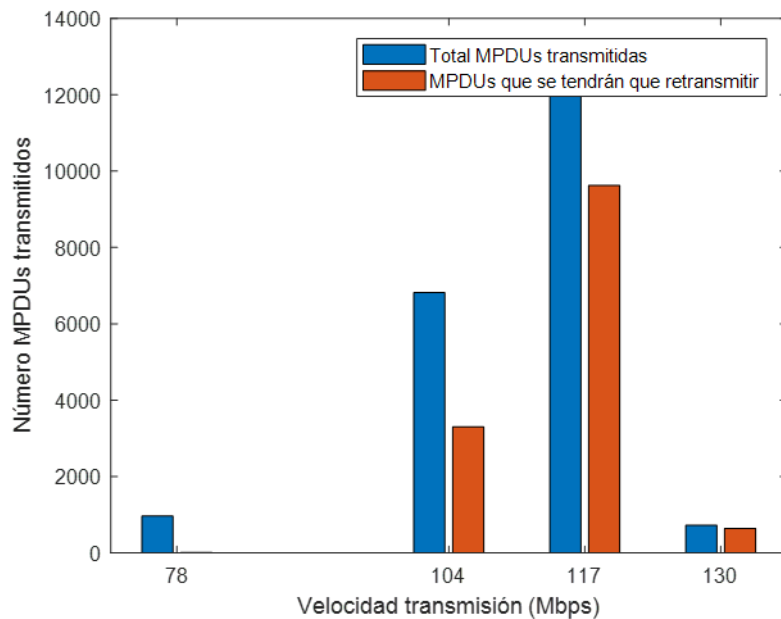
Para la representación de esta gráfica se empleará un histograma, usando el comando “bar”. Puede haber dos casos posibles de representación de la gráfica. El primero que es el normal en el cual aparecen varias velocidades de transmisión, y el segundo, que es el caso raro, en el que solo aparece una velocidad de transmisión.

```

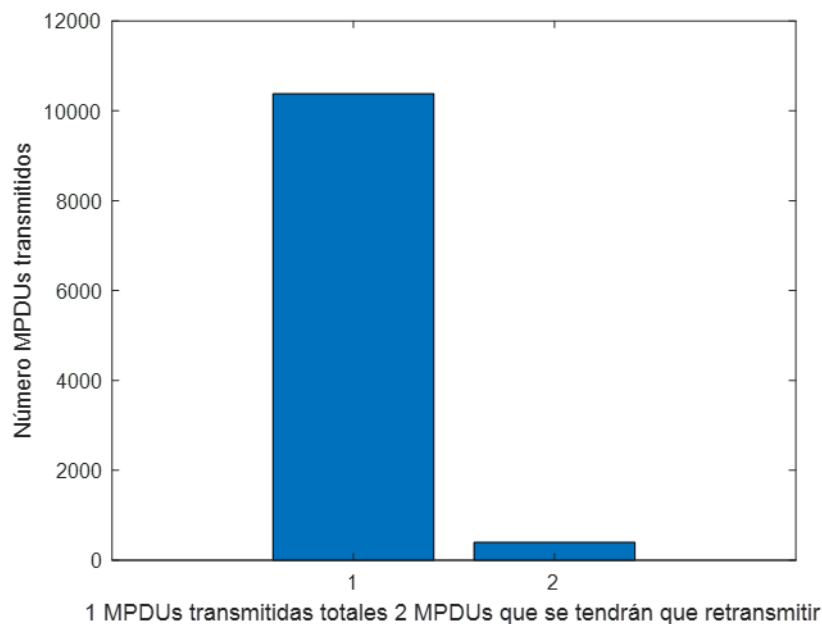
figure(4)
xbins=Sol(:,1);
ybins=Sol(:,2:3);
tam=size(xbins);
tam=tam(1,1)
if tam > 1
    bar(xbins,ybins)
    xlabel('Velocidad transmisión (Mbps)')
    ylabel('Número MPDUs transmitidos')
    legend('Total MPDUs transmitidas','MPDUs que se tendrán que retransmitir')
else
    bar(ybins)
    xlabel('1 MPDUs transmitidas totales 2 MPDUs que se tendrán que retransmitir')
    ylabel('Número MPDUs transmitidos')
end

```

*Figura 5.37. Bloque de código para la representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad*



*Figura 5.38. Representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad*



*Figura 5.39. Representación del número de tramas transmitidas y cantidad de ellas que se retransmitirán a una única velocidad*

La cuarta gráfica que se representará será la probabilidad de retransmisión de una trama teniendo en cuenta su velocidad de transmisión. Para ello se emplearán los valores de la matriz Sol. Se ordenarán los valores de la matriz Sol según su velocidad de transmisión con el comando "sortrows". Luego se creará la matriz Prtx de dos columnas y mismas filas que la matriz Sol. La primera columna de Prtx contendrá las velocidades de transmisión, es decir, será idéntica a la primera columna de la matriz Sol mientras que la segunda columna representará la probabilidad de retransmisión para cada velocidad de transmisión. Esto se calcula dividiendo el número de

tramas a esa velocidad de transmisión que se van a retransmitir frente al número total de trama transmitidas. Esto supone la división de la tercera columna entre la segunda columna de la matriz Sol.

```
Sol = Sol(1:Rateunico-1,:);
Sol = sortrows(Sol);
Prtx = zeros(Rateunico-1,2);

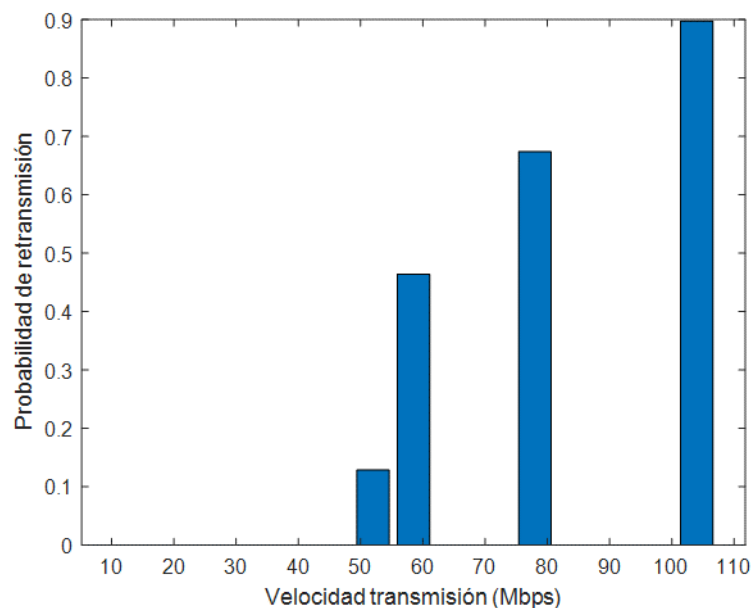
for i = 1:size(Sol)
    Prtx(i,1)=Sol(i,1);
    Prtx(i,2)=Sol(i,3)/Sol(i,2);
end
```

*Figura 5.40. Bloque de código para la obtención de la probabilidad de transmisión según la velocidad de transmisión*

Para la representación de la gráfica se empleará un histograma, utilizando el comando “bar”.

```
figure(3)
xbins=Prtx(:,1);
ybins=Prtx(:,2);
bar(xbins,ybins)
xlabel('Velocidad transmisión (Mbps)')
ylabel('Probabilidad de retransmisión')
```

*Figura 5.41. Bloque de código para la representación de la probabilidad de transmisión según la velocidad de transmisión*



*Figura 5.42. Representación de la probabilidad de transmisión según la velocidad de transmisión*

La quinta y última gráfica que se representará será el número de paquetes que se retransmiten un determinado número de veces. Para ello se utilizarán los datos obtenidos del archivo de texto "Relaciones.txt". Para ello primero se realizará un ciclo "for" similar al del código de la representación de la tercera gráfica. Este ciclo "for" servirá para indicar cuantas veces se transmite cada trama. El ciclo "for" irá leyendo datos de la matriz C (datos de "Relaciones.txt"), y entrará en una condición tipo "if" si la información leída de la trama detalla que se va a retransmitir (tercera matriz columna C) o si es una "retransmisión fantasma" (quinta columna matriz C). Una vez pasada la condición, habrá otro ciclo "for" que comparará el identificador de la trama con una lista contenida en la primera columna de la matriz Solrepe. Si el identificador aparece ya anteriormente en esa lista significa que esa trama ya se ha transmitido anteriormente por lo que ha habido retransmisiones e indica que es repetida. La segunda columna de la matriz Solrepe indica las veces que se ha transmitido una trama en particular, por lo que, si es una trama repetida, se tendrá que incrementar el valor en 1 de esa columna respecto a la fila que contiene el identificador de la trama.

Una vez salido del ciclo "for", se entrará en una condición "if" en la que si la trama es la primera vez que aparece, es decir, no es repetida, se añadirá una nueva fila en la matriz Solrepe con el identificador de la trama, y en la segunda columna se sumará 1 si se va a retransmitir y también se sumará 1 si es una "retransmisión fantasma", recordando que una "retransmisión fantasma" es aquella trama que aparece por primera vez, pero es una retransmisión.

```

for i= 1:maximo
    if C(i,3) == 1 || C(i,5) == 1
        for j = 1: Contrepe
            if C(i,1)== Solrepe(j,1)
                Solrepe(j,2) = Solrepe(j,2) +1;
                repe=1;
            end
        end
        if repe == 0
            Solrepe(j,1) = C(i,1);
            Solrepe(j,2) = C(i,3) + C(i,5);
            Contrepe=Contrepe +1;
        else
            repe=0;
        end
    end
end
end

```

*Figura 5.43. Bloque de código para la obtención del número de veces que se retransmite cada trama*

Por último, una vez completado el bloque de código anterior, utilizando los datos de la matriz Solrepe, se creará la matriz ContadorRTX, cuya primera columna indicará el número de veces que se habrá retransmitido y la segunda columna indicará el número de paquetes que habrá sufrido ese número de retransmisiones. Para ello se empleará un ciclo "for".

```

for i = 1:M
    ContadorRTX(i,1) = i;
    for j = 1:Maximo2
        if ContadorRTX(i,1) == Solrepe(j,2)
            ContadorRTX(i,2)=ContadorRTX(i,2)+1;
        end
    end
end
end

```

Figura 5.44. Bloque de código para la obtención del número de veces que se transmite una retransmisión

Para la representación de la gráfica se empleará un histograma, utilizando el comando “bar”.

```

figure(5)
xbins=ContadorRTX(:,1);
ybins=ContadorRTX(:,2);
bar(xbins,ybins)
ylabel('MPDUs retransmitidas')
xlabel('Número de retransmisiones por MPDU')

```

Figura 5.45. Bloque de código para la representación del número de veces que se transmite una retransmisión

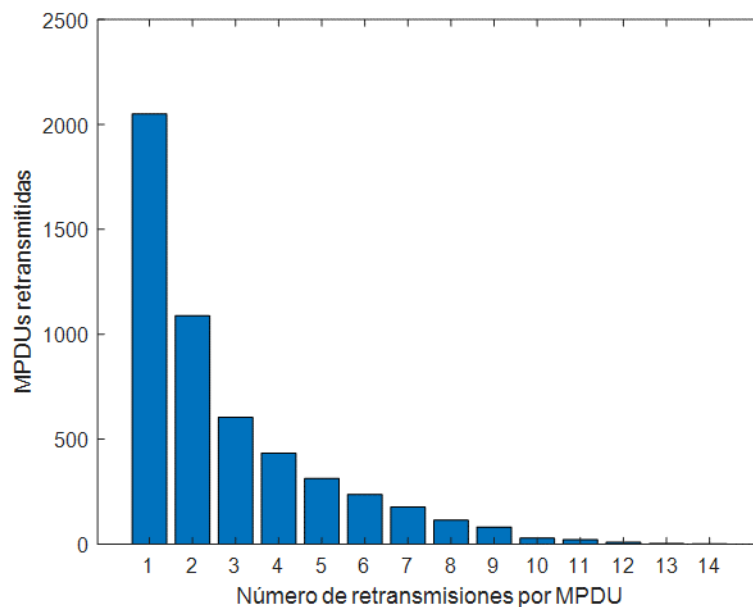


Figura 5.46. Representación del número de veces que se transmite una retransmisión

## 5.5 Medidas con Iperf

Una vez realizadas las medidas con Nttcp, se realizarán varias medidas con Iperf, ya que nos puede proporcionar más datos diferentes a los de Nttcp como por ejemplo el Jitter. La

principal diferencia con Nttcp es que en vez de indicarle un buffer de bytes que se deben transmitir hasta haberse enviado al completo, en Iperf se indica una cantidad de tiempo y en ese tiempo se envían todos los bytes posibles. Para resumir, en Nttcp se indica cuántos bytes se tienen que enviar en el tiempo que sea mientras que en Iperf indicado un tiempo por el usuario se debe de enviar el máximo posible de bytes en ese espacio de tiempo.

En este caso se realizará primero una medida Nttcp para ver cuánto se tarda en enviarse 10000 paquetes de 1472 Bytes de datos más cabeceras. Una vez obtenido ese tiempo se utilizará en Iperf para que nos dé un valor aproximado correcto. Las medidas se realizarán en el escenario 1 por la mañana, que es el mejor caso posible, y en el escenario 2 por la tarde, que es el peor caso posible, teóricamente.

La medida nttcp, interfaz de red en modo monitor, filtrado, obtención de datos y representación gráfica se realiza como en los pasos anteriores por lo que solo se explicará en este apartado como se realizan las medidas Iperf.

### ➤ Receptor

En el ordenador receptor se abrirá un terminal y se escribirá la siguiente línea de comando la cual se explicarán sus parámetros, para que el ordenador receptor esté listo para recibir la transmisión de datos:

```
iperf -s -u -l 1472
```

-s = indica que realiza la función de servidor, es decir, se encargará de recibir los paquetes

-u = se recibirá tráfico UDP

-l = indica la longitud del buffer

### ➤ Transmisor

En el ordenador transmisor, una vez el ordenador receptor esté listo para recibir la transmisión de paquetes, se abrirá una terminal y se introducirá la siguiente línea de comando y se explicará a continuación que significan todos los parámetros que contiene:

```
iperf -c "dirección IP destino" -u -b 100M -l 1472 -t 25 -i 1
```

-c = indica que realiza la función de cliente, es decir, se encarga de transmitir los paquetes

"dirección IP destino" = dirección IP del UDP

-b = ancho de banda a utilizar en bits/seg, en nuestro caso asegurándonos de poner un valor elevado

-l = indica la longitud del buffer

-t = indica la duración del análisis. Este valor se obtiene de la medida nttcp

-i = indica los segundos entre los reportes de ancho de banda periódicos

Una vez ejecutada esta línea de comando, se producirá la transmisión de paquetes, y se verá como cada tiempo indicado en -i nos aparecerá un reporte y el reporte final del ordenador receptor en el ordenador transmisor, mientras que en el ordenador receptor solo nos aparecerá un único reporte con resultados de la medida.

### ➤ Resultados

```
rga91@Rodrigo:~$ iperf -s -u -l 1472
-----
Server listening on UDP port 5001
Receiving 1472 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 192.168.1.104 port 5001 connected with 192.168.1.41 port 39557
[ ID] Interval      Transfer    Bandwidth    Jitter    Lost/Total Datagrams
[ 3] 0.0- 7.2 sec  11.3 MBytes 13.2 Mbits/sec 3.998 ms   78/ 8113 (0.96%)
```

*Figura 5.47. Resultados de Iperf en el ordenador receptor*

En la imagen que pertenece al ordenador receptor se puede ver el reporte final de la transferencia de paquetes con los resultados extraídos de ella. En este reporte aparece cuanto ha durado la prueba, que cantidad de datos han sido recibidos, el ancho de banda medio de la transmisión, el jitter y la cantidad de paquetes perdidos, paquetes totales enviado y el PER (Packet error rate).

```
rodrigo@rodrigo-Aspire-A315-21:~$ iperf -c 192.168.1.104 -u -b 100M -l 1472 -t 7
.14 -i 1
-----
Client connecting to 192.168.1.104, UDP port 5001
Sending 1472 byte datagrams, IPG target: 112.30 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 3] local 192.168.1.41 port 39557 connected with 192.168.1.104 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 1.0 sec  1.73 MBytes 14.5 Mbits/sec
[ 3] 1.0- 2.0 sec  1.56 MBytes 13.1 Mbits/sec
[ 3] 2.0- 3.0 sec  1.74 MBytes 14.6 Mbits/sec
[ 3] 3.0- 4.0 sec  1.57 MBytes 13.1 Mbits/sec
[ 3] 4.0- 5.0 sec  1.86 MBytes 15.6 Mbits/sec
[ 3] 5.0- 6.0 sec  1.59 MBytes 13.3 Mbits/sec
[ 3] 6.0- 7.0 sec  1.14 MBytes 9.53 Mbits/sec
[ 3] 0.0- 7.2 sec  11.4 MBytes 13.3 Mbits/sec
[ 3] Sent 8113 datagrams
[ 3] Server Report:
[ 3] 0.0- 7.2 sec  11.3 MBytes 13.2 Mbits/sec 0.000 ms 78/ 8113 (0%)
```

*Figura 5.48. Resultados de Iperf en el ordenador transmisor*

En la imagen que pertenece al ordenador transmisor se puede ver como cada segundo va ofreciendo un reporte con la transferencia de Bytes y el ancho de banda en ese lapso. El último reporte corresponde al de toda la transferencia y deberá ser similar al reporte del servidor que se ve en el borde inferior de la imagen. Por último, justo encima de “Server report”, indica cuantos paquetes han sido enviados.



## Capítulo 6. Análisis de los resultados

---

En este apartado se analizarán todos los datos obtenidos de las medidas y se razonará el porqué de esos resultados, dependiendo de la estructura de red empleada a la realización de las medidas: Ad-hoc e infraestructura

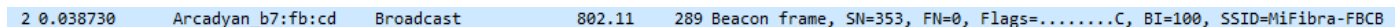
### 6.1 Medidas en modo infraestructura

En este apartado se analizarán las medidas realizadas en una red Wi-Fi en modo infraestructura, es decir, con un router como intermediario por lo que habrá un enlace de subida (Transmisor → Router) y un enlace de bajada (Router → Receptor). Para ello habrá cuatro subapartados, cada uno correspondiente al escenario empleado según la posición del transmisor y el horario en el que se realizó las medidas.

En la realización de este conjunto de medidas se empleará una red Wi-Fi en modo infraestructura, que como se mencionó en la Introducción en el punto 1.1 Presentación y objetivos, consiste en una red Wi-Fi en la cual los dispositivos de la red Wi-Fi están conectados a un punto de acceso (Router) y este punto de acceso estará conectado a una red cableada lo cual permitirá la conexión a Internet. En este apartado se detallarán las medidas y explicación de los resultados obtenidos en red Wi-Fi en modo infraestructura en un entorno doméstico.

Antes de comenzar a analizar las medidas se analizará una trama de “beacon”. Las tramas beacon de una red Wi-Fi contienen toda la información relacionada con la red inalámbrica y son transmitidos periódicamente (valor definido por el Beacon Interval) para anunciar su presencia en un canal determinado. Nuestro dispositivo irá cambiando de canal y escuchará durante un tiempo determinado todos los beacons posibles. Este método de búsqueda de redes se denomina pasivo. En este caso, los beacons serán transmitidos por el punto de acceso, el router, ya que se trabaja en este caso en modo infraestructura.

Para visualizar una trama de beacon se abrirá cualquier captura de una medida en Wireshark y buscará una trama que el campo de información indique que es “Beacon frame” y que el SSID (nombre de la red Wi-Fi) sea el de nuestra red Wi-Fi.



*Figura 6.1. Formato trama Beacon en Wireshark*

La trama beacon se transmitirá a la mínima velocidad posible (1 Mbps) utilizando el estándar 802.11b, ya que como se dijo en el capítulo 2 los estándares 802.11b, 802.11g y 802.11n son compatibles, y el más restrictivo en cuanto características es el 802.11b. Esto permite conectarse a un mayor número de dispositivos que empleen cualquiera de esos tres estándares a la red Wi-Fi.

El campo que nos interesará de la trama Beacon es aquel donde se menciona las características de la red: IEEE 802.11 Wireless LAN. En este campo se podrá encontrar características básicas como velocidades de transmisión básicas soportadas (estándares 802.11b y 802.11g), tipo de seguridad que emplea, nombre de la red inalámbrica (SSID) y en que canal

se encuentra. También se puede encontrar información de la regulación de nuestro país aquí, mencionado ya en el capítulo 2, como que solo hay 13 canales y el canal 14 no se usa y que la potencia de transmisión máxima es 30 dBm (100 mW):

*Figura 6.2. Campo información sobre regulación del país*

```

▼ Tag: Country Information: Country Code ES, Environment Any
  Tag Number: Country Information (7)
  Tag length: 6
  Code: ES
  Environment: Any (0x20)
  > Country Info: First Channel Number: 1, Number of Channels: 13, Maximum Transmit Power Level: 30 dBm

```

Ya que se va a trabajar usando el estándar 802.11n, nos interesarán en especial dos campos, HT Capabilities (802.11n D1.10) y HT Information (802.11n D1.10), poniendo especial énfasis en el primero en que se muestra que características de 802.11n ofrece el router.

```

▼ HT Capabilities Info: 0x09ac
  .... = HT LDPC coding capability: Transmitter does not support receiving LDPC coded packets
  .... = HT Support channel width: Transmitter only supports 20MHz operation
  .... 11.. = HT SM Power Save: SM Power Save disabled (0x3)
  .... = HT Green Field: Transmitter is not able to receive PPDU's with Green Field (GF) preamble
  .... 1.. = HT Short GI for 20MHz: Supported
  .... 0.. = HT Short GI for 40MHz: Not supported
  .... 1... = HT Tx STBC: Supported
  .... 01... = HT Rx STBC: Rx support of one spatial stream (0x1)
  .... 0... = HT Delayed Block ACK: Transmitter does not support HT-Delayed BlockAck
  .... 1... = HT Max A-MSDU length: 7935 bytes
  .... 0... = HT DSSS/CCK mode in 40MHz: Won't/Can't use of DSSS/CCK in 40 MHz
  .... 0... = HT PSMP Support: Won't/Can't support PSMP operation
  .... 0... = HT Forty MHz Intolerant: Use of 40 MHz transmissions unrestricted/allowed
  .... 0... = HT L-SIG TXOP Protection support: Not supported

```

*Figura 6.3. Características opcionales de 802.11n que ofrece mi router*

El adaptador Wi-Fi USB una vez se conecte al ordenador y tenga esté la conexión Wi-Fi habilitada comenzará a enviar una trama “Probe request” con información sobre el adaptador Wi-Fi USB y sus características en cada uno de los canales, y este “Probe request” será respondido por todos los puntos de acceso que lo reciban con un “Probe response” (que es muy similar a una trama beacon). Este método de búsqueda de redes se denomina activo.

Al igual que antes, en la trama “Probe request” se la identificará por el campo información lleva incluido que es una trama “Probe request” y por el origen de la trama gracias a que ya se conoce el nombre del dispositivo:

```

384 13.129690  Tp-LinkT_21:08:f2  Broadcast  802.11  254 Probe Request, SN=1132, FN=0, Flags=.....C, SSID=wildcard (Broadcast)

```

*Figura 6.4. Formato trama “probe request” del dispositivo TP-Link*

La trama “Probe Request” al igual que la trama “Beacon” mencionada anteriormente son muy similares en estructura. Tendrán muchos campos similares en lo que único que se diferenciará serán las características de cada dispositivo. Al igual que antes se va a hacer especial énfasis en el campo HT Capabilities (802.11n D1.10) que es donde se explican las características que ofrece el adaptador Wi-Fi USB sobre el estándar 802.11n.

```

▼ HT Capabilities Info: 0x11ee
.... ..0 = HT LDPC coding capability: Transmitter does not support receiving LDPC coded packets
.... ..1 = HT Support channel width: Transmitter supports 20MHz and 40MHz operation
.... ..11.. = HT SM Power Save: SM Power Save disabled (0x3)
.... ..0 .... = HT Green Field: Transmitter is not able to receive PPDU with Green Field (GF) preamble
.... ..1. .... = HT Short GI for 20MHz: Supported
.... ..1.. .... = HT Short GI for 40MHz: Supported
.... ..1... .... = HT Tx STBC: Supported
.... ..01 .... = HT Rx STBC: Rx support of one spatial stream (0x1)
.... ..0... .... = HT Delayed Block ACK: Transmitter does not support HT-Delayed BlockAck
.... ..0... .... = HT Max A-MSDU length: 3839 bytes
.... ..1 .... = HT DSSS/CCK mode in 40MHz: Will/Can use DSSS/CCK in 40 MHz
.... ..0. .... = HT PSMP Support: Won't/Can't support PSMP operation
.... ..0.. .... = HT Forty MHz Intolerant: Use of 40 MHz transmissions unrestricted/allowed
.... ..0... .... = HT L-SIG TXOP Protection support: Not supported

```

*Figura 6.5. Características opcionales de 802.11n que ofrece el adaptador Wi-Fi USB TL-WN821N*

### 6.1.1 Análisis medidas escenario 1

Se han realizado 5 medidas en este escenario, en el cual tanto el transmisor como el Router están juntos, alejados del receptor. Para representar los datos obtenidos de las medidas Nttcp se realizará una tabla que representará los datos obtenidos de las 5 medidas. La información que se representará será el Throughput (Mbps), el PER (Packet Error Rate = probabilidad de error en la transmisión de un paquete) que sabiendo que en la realización de las medidas se envían 10000 paquetes con el dato del PER se podrá obtener el número de paquetes perdidos y por último otros datos que se representarán tanto para el enlace de subida como de bajada serán el número total de tramas procesadas, el número de tramas retransmitidas y el FER (Frame Error rate= probabilidad de error de una trama), que se obtiene de la división entre los datos mencionados anteriormente:

$$FER = \frac{\text{Tramas retransmitidas}}{\text{Tramas procesadas}}$$

Para el PER seguirá una fórmula similar, pero esta vez con una división entre los paquetes perdidos y los paquetes totales transmitidos:

$$PER = \frac{\text{Paquetes perdidos}}{\text{Paquetes totales transmitidos}}$$

Como se podrá comprobar algunos de estos datos se podrán obtener a partir de otros datos como puede ser el FER o el PER, mientras que otros datos se obtendrán por Matlab, como respuesta de la herramienta de medidas Nttcp cuando finaliza una medida o simplemente como valor de entrada que el usuario elige como por ejemplo el número de paquetes que se desea transmitir, en este caso 10000 paquetes.

Los datos que nos devuelve la herramienta de medida ya se mencionaron en el apartado 5.2.3, pero solo se tendrán en cuenta para la realización de las tablas dos que serán el valor correspondiente a la segunda fila y en la columna “Real-Mbit/s” que se corresponderá con el Throughput (Mbps) y el segundo valor que se tomará en cuenta será el de la columna “Calls” en el cual el valor de la primera fila deberá ser siempre 10003 y el de la segunda fila que es del cual se obtendrá el número de paquetes será igual o inferior a 10001 en nuestro caso. Con este valor se podrá calcular el número de paquetes perdidos ya que cuando el valor indica 10001 no se habrá perdido ninguno mientras que si es inferior la diferencia entre el número obtenido y 10001 será el número de paquetes perdidos:

$$\text{Paquetes Perdidos} = 10001 - \text{Valor obtenido de Nttcp}$$

El número de paquetes perdidos también se podría obtener empleando el valor que devuelve la primera columna de la respuesta de la herramienta de medidas Nttcp, que es “Bytes”. La primera fila indica los bytes enviados mientras que la segunda indica los bytes finalmente recibidos. Con esto, sabiendo el número de paquetes totales transmitidos y el tamaño de cada paquete en la zona de datos que se indica al ejecutar la herramienta de medida Nttcp, en este caso 1472, se podrá obtener también el número paquetes perdidos:

$$\text{Paquetes Perdidos} = \frac{\text{Bytes totales Transmitidos} - \text{Bytes totales recibidos}}{\text{Tamaño Campo Datos por paquete}}$$

Para los datos de Tramas procesadas y Tramas retransmitidas se empleará el programa Matlab. Estos datos simplemente representan el número total de tramas que ha capturado el adaptador Wi-Fi USB en modo monitor (Tramas procesadas) y cuantas de estas tramas son retransmisiones (Tramas retransmitidas). Cabe destacar que debido a que el “Sniffer” (adaptado Wi-Fi USB en modo monitor) no captura todas las tramas posibles este valor no es exacto, y habrá varios métodos para obtener valores aproximados. En mi caso, se ha tomado los valores de la figura 5.38 que se encuentra en el apartado 5.4, en el cual se representa básicamente el número total de tramas procesada por velocidad y cuáles de ellas se van a retransmitir. Haciendo un sumatorio de esos valores obtengo los valores de Tramas Procesadas y Tramas Retransmitidas. Aplico este método ya que es el más fiel si se tiene en cuenta solamente la captura realizada. Los demás métodos para obtener aproximadamente estos valores pasan por ver como se considera las “retransmisiones fantasmas”, aclarando que aplicando cualquier método solo hay como mucho una diferencia del 1% en los peores casos en los valores, ya que las “retransmisiones fantasmas” son muy escasas.

Por último, una vez explicada la tabla de las medidas Nttcp y la procedencia de los datos que aparecen en ella, se añadirá otra tabla que contendrá la media y la varianza de los valores que se encuentra en la tabla de las medidas Nttcp.

Para calcular la media se sumarán todos los valores sobre los cuales se desea obtener la media y se dividirá entre el número de valores sobre los que se obtendrá la media:

$$Media = \frac{(X_1 + X_2 + \dots + X_N)}{N}$$

Para calcular la varianza se empleará la media obtenida anteriormente y aplicando la siguiente formula se obtendrá el valor correspondiente:

$$Varianza = \frac{(X_1^2 + X_2^2 + \dots + X_N^2)}{N} - Media^2$$

Una vez explicado todo esto, se mostrará la primera tabla perteneciente al escenario 1:

*Tabla 6.1. Escenario 1: Transmisor y Router juntos por la mañana en una red Wi-Fi en modo infraestructura*

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas Enlace subida	Tramas RTX Enlace subida	FER Enlace subida (%)	Tramas procesadas Enlace bajada	Tramas RTX Enlace bajada	FER Enlace bajada (%)
1	30,06	0,16	11753	2448	20,82	10942	1107	10,11
2	32,07	0	9955	82	0,82	10504	576	5,48
3	36,71	0,02	9987	144	1,44	10126	398	3,93
4	19,44	0,05	9015	171	1,89	10250	975	9,51
5	26,64	0,05	9742	348	3,57	9997	621	6,21

Media	28,984	0,056	10090	638,6	5,708	10364	735,4	7,048
Varianza	33,3909	0,0031	814230	826290	57,9258	111600	69566	5,664

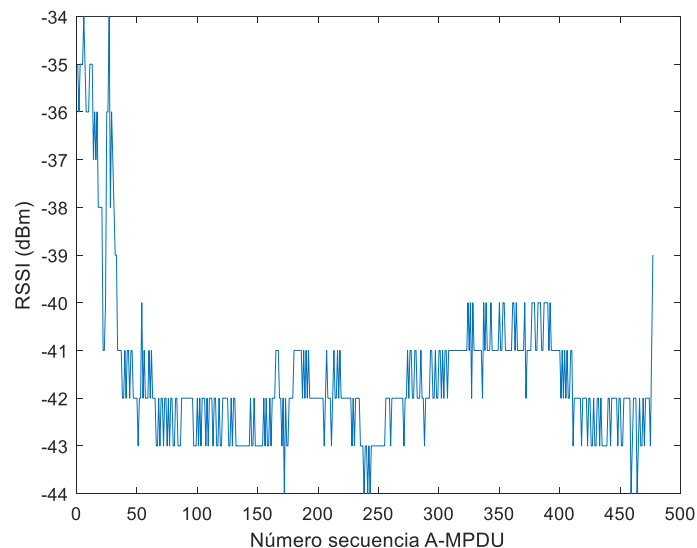
En la tabla mostrada se puede comprobar que la pérdida de paquetes es siempre inferior al 1% siendo una conexión muy buena. Cabe recordar que el throughput mostrado representa al conjunto del enlace de subida como el enlace de bajada, teniendo una media 28,94 Mbps, recordando que, aunque el flujo de tráfico entre el router y portátil transmisor sea mínimo, el enlace de bajada siempre tendrá que recorrer la misma distancia para llegar al receptor, por lo que la calidad del enlace de bajada en teoría debería ser de peores características (o iguales dependiendo del escenario) que el enlace de subida. Esto sería aplicable si ambos dispositivos, Router y adaptador Wi-Fi USB tuvieran los mismos mecanismos de retransmisión que se explicarán más adelante en 6.1.2.

Como cabría esperar, el FER del enlace de subida es menor que el del enlace de bajada aun habiendo un pico en la medida de empeoramiento de la calidad de transmisión en la medida 1 en el enlace de subida. El FER es inferior al 8% lo cual dice que habido pocas retransmisiones y junto al valor del PER dan a indicar que la conexión es de una calidad buena.

Ahora se cogerá una medida, en este caso la medida 5, y se representará información relacionada con esta y explicará tanto para el enlace de subida como de bajada.

#### ➤ Enlace subida

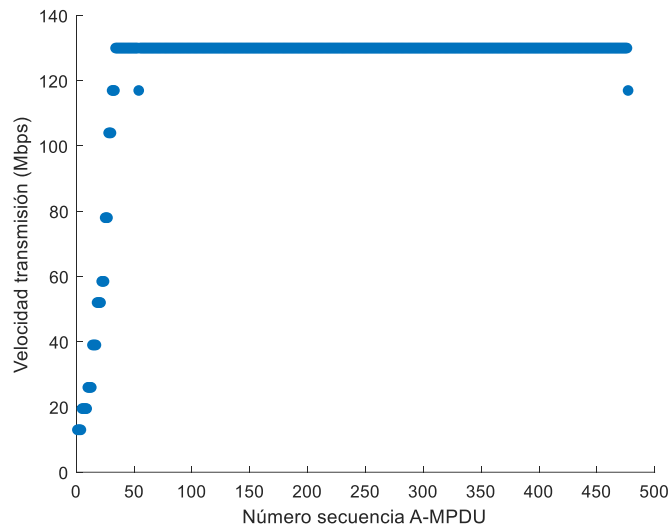
La primera gráfica representará la potencia de señal recibida en el transmisor que es donde se captura la información. Todas las figuras obtenidas que traten sobre la potencia de señal recibida en el transmisor en el enlace de subida se podrían traducir como una representación de la potencia de señal emitida por el transmisor.



*Figura 6.6. Potencia de señal recibida en el transmisor en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura*

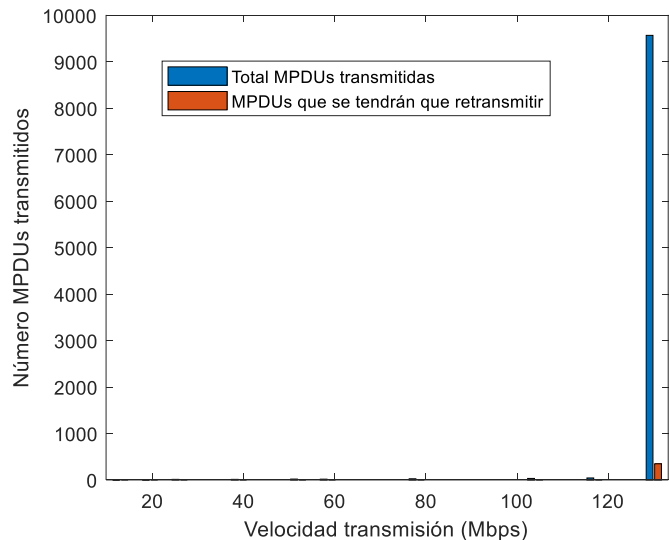
La segunda gráfica representa la velocidad de transmisión empleada en cada A-MPDU. Como se podrá observar, al inicio de la transmisión se produce incremento en poco tiempo de la velocidad de transmisión poco a poco hasta llegar a una velocidad de transmisión estable de

130 Mbps en este caso. Esto es debido que para la realización de las capturas lo primero de todo es poner a capturar Wireshark y luego justo después unirse a la red Wi-Fi para capturar las tramas EAPOL como ya se mencionó para luego poder decodificar los datos. Por eso, debido a la reciente unión a la red Wi-Fi se va incrementando poco a poco la velocidad de transmisión (incremento MCS index), comenzando incluso en velocidades correspondientes a 802.11b, hasta llegar a un valor adecuado como es este caso 130 Mbps.



*Figura 6.7. Velocidad de transmisión en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura*

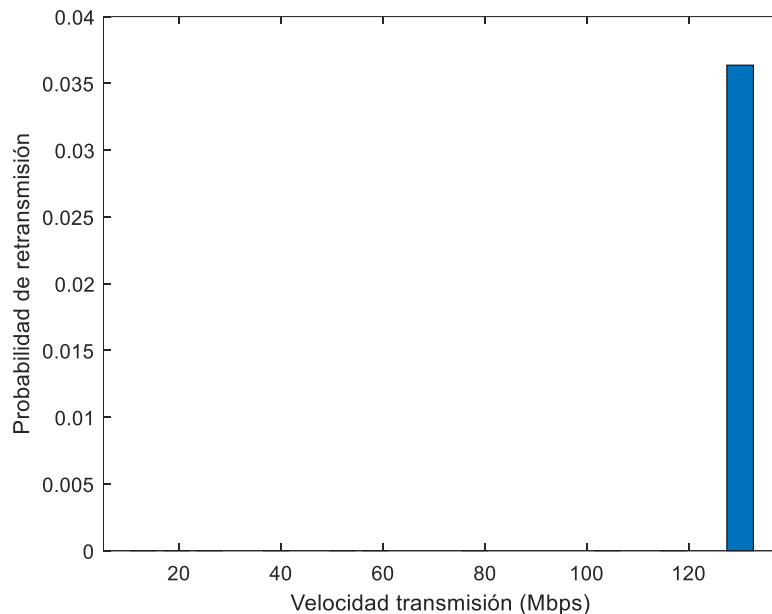
La siguiente gráfica representa las tramas que se transmiten a una velocidad y cuantas de esas tramas se retransmitirán debido a cualquier motivo, ya sea a esa misma velocidad o a otra. Como se puede comprobar que, aunque en la gráfica anterior se aprecian otras velocidades de transmisión, en esta gráfica se verifica que representan en cuanto a cantidad de tramas un número ínfimo, siendo la velocidad estándar para las tramas 130 Mbps como se mencionó. También se puede apreciar el bajo el nivel de retransmisiones que se produce.



*Figura 6.8. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura*



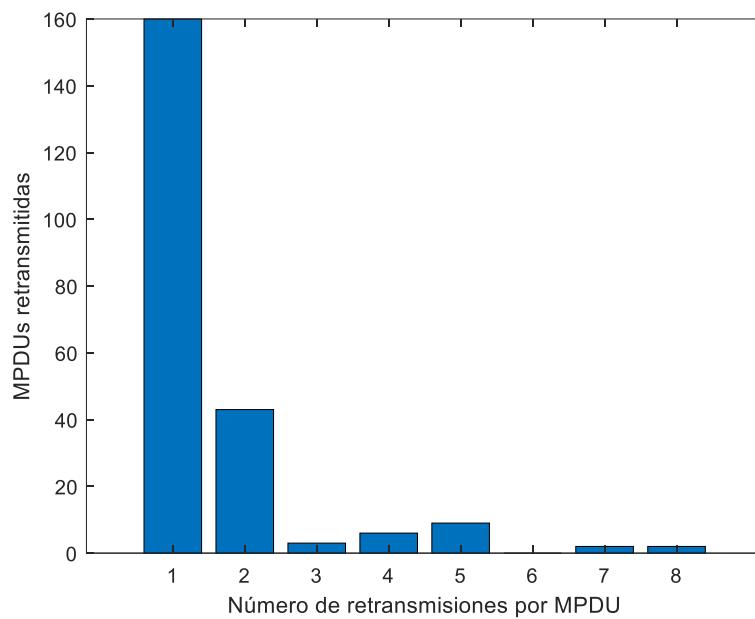
En la próxima gráfica, se mostrará la probabilidad de retransmisión de una trama si transmite a una determinada velocidad. En este caso los resultados son muy buenos representando la probabilidad de retransmisión inferior al 4% (0,04 sobre 1) para la velocidad de transmisión de los 130 Mbps que es la velocidad a la que van casi todas las tramas en esta medida.



*Figura 6.9. Probabilidad de retransmisión a una determinada velocidad en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura*

A veces hay casos donde se muestran en esta última gráfica datos dispares como por ejemplo velocidades de transmisión bajas con una mayor probabilidad de retransmisión que una velocidad de transmisión alta. Esto es debido a que hay veces debido a un empeoramiento de la calidad de la conexión la velocidad de transmisión en un momento dado se reduce para poder transmitir las tramas, pero aun así las tramas se pierden, quedando por ejemplo 10 tramas transmitidas a esa velocidad, 8 perdidas, por lo que la probabilidad de retransmisión sería alta. Por eso es importante observar ambas gráficas juntas, ya que podría manifestarse una probabilidad de transmisión muy alta en la figura 6.9, pero en la figura 6.8, apreciarse que el número de tramas es insignificante como para tenerlo en cuenta. Esto se podría resolver con un filtro que solo representará tramas de una misma velocidad de transmisión a partir de que lleguen a una cantidad determinada. En mi caso, no he aplicado este filtro.

La última gráfica que se mostrará corresponderá con el número de retransmisiones que se retransmite una retransmisión, es decir, en el valor 1 del eje x indicará que una cantidad de aproximadamente 160 tramas que son retransmisiones solo se transmiten 1 vez, mientras que otras retransmisiones se producirán más veces.

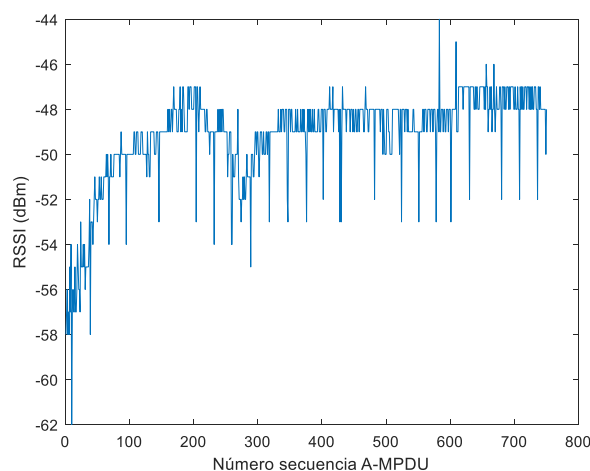


*Figura 6.10. Número de veces que se transmite una retransmisión en el escenario 1, enlace de subida, en una red Wi-Fi en modo infraestructura*

Como se verá más adelante en 6.1.2, se explicará el mecanismo de retransmisión del adaptador Wi-Fi USB, que explicará por qué el número de retransmisiones y elección de velocidades difiere del empleado por el Router. No se explicará en este escenario ya que donde mejor se podrá comprobar estas diferencias es cuando ambos receptor y transmisor estén juntos alejados del Router.

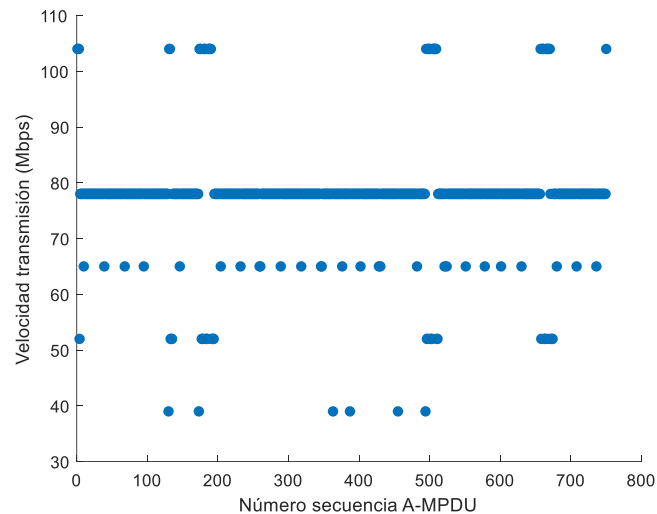
#### ➤ Enlace bajada

Las gráficas que se representarán mostrarán la misma información, pero en lo relativo al enlace de bajada.



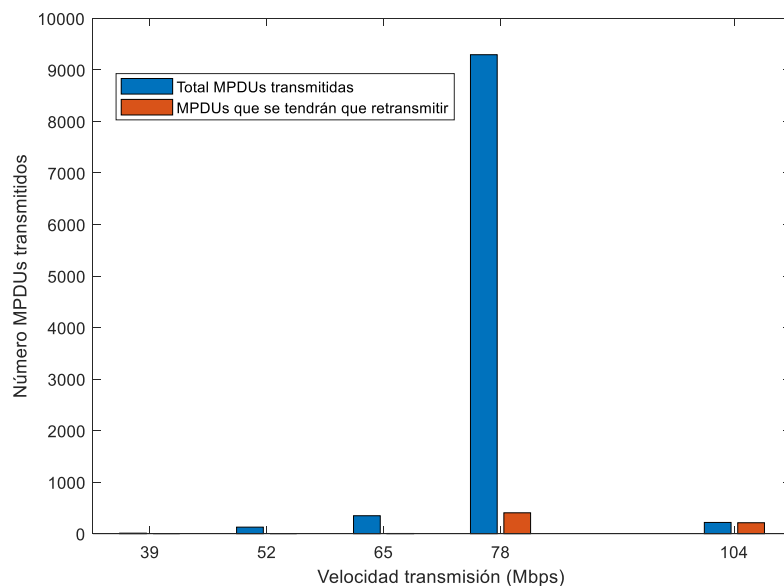
*Figura 6.11. Potencia de señal recibida en el transmisor en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura*

La figura 6.11 representa la potencia de señal con la que transmite el router captada en el transmisor. Esto es posible debido a que el transmisor y el receptor están al lado. Para saber la potencia de señal recibida en el receptor se mostrará en el enlace de bajada de los escenarios donde tanto el receptor y el transmisor estén juntos alejados del router y que será igual para los escenarios que se realicen en el mismo horario ya que el enlace de bajada siempre tendrá la misma distancia (Ver Figura 6.21).

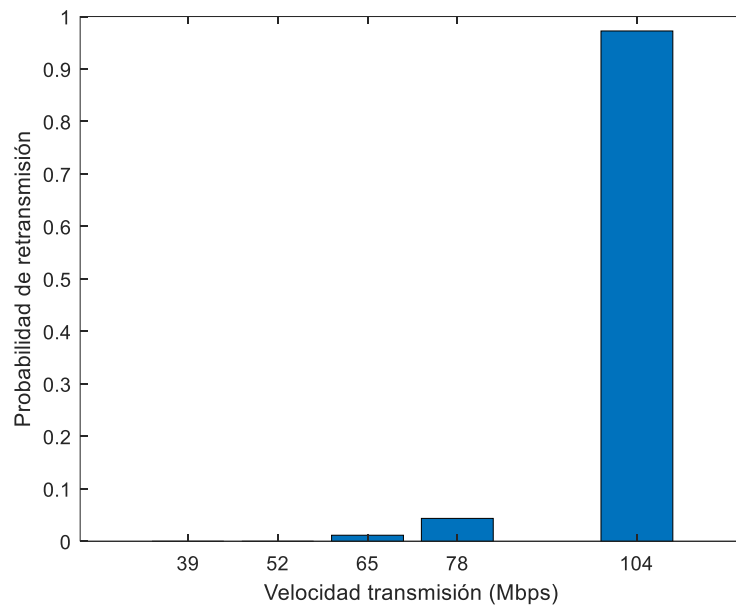


*Figura 6.12. Velocidad de transmisión en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura*

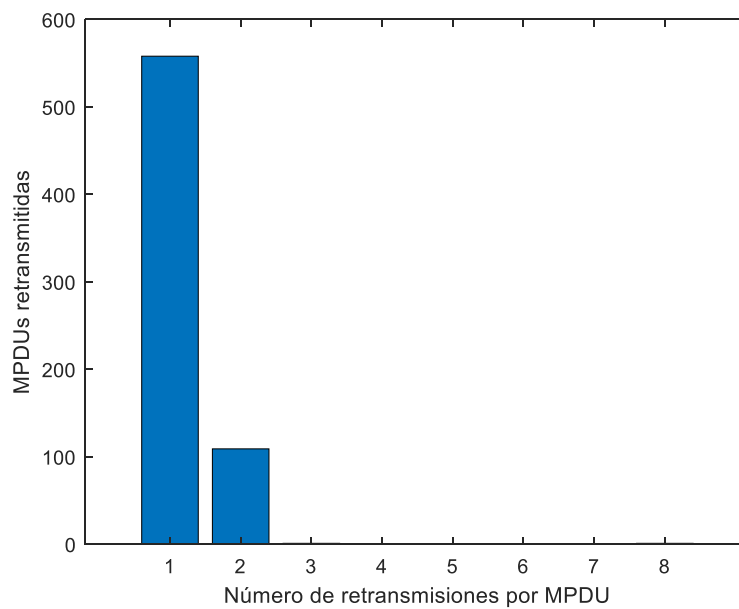
En esta gráfica se puede apreciar un cambio constante en las velocidades debido a la mayor distancia que hay n el enlace de subida, al mecanismo de retransmisión y al método de elección de velocidades que emplea el router (Ver 6.1.2).



*Figura 6.13. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.14. Probabilidad de retransmisión a una determinada velocidad en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.15. Número de veces que se transmite una retransmisión en el escenario 1, enlace de bajada, en una red Wi-Fi en modo infraestructura*

Como se verá más adelante en 6.1.2 Enlace subida, se explicará el mecanismo de retransmisión del Router, que explicará por qué el número de retransmisiones y elección de velocidades difiere del empleado por el adaptador Wi-Fi USB. No se explicará en este escenario ya que donde mejor se podrá comprobar estas diferencias es cuando ambos receptor y transmisor estén juntos alejados del Router.

### **6.1.2 Análisis medidas escenario 2**

Se han realizado 5 medidas en este escenario, en el cual tanto el transmisor como el receptor están juntos, alejados del Router. Para representar los datos obtenidos de las medidas Nttcp se realizará una tabla que representará los datos obtenidos de las 5 medidas como la empleada anteriormente (Ver 6.1.1 para consultar información acerca de la tabla):

Tabla 6.2. Escenario 2: Transmisor y Receptor juntos por la mañana alejados del router en una red Wi-Fi en modo infraestructura

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas Enlace subida	Tramas RTX Enlace subida	FER Enlace subida (%)	Tramas procesadas Enlace bajada	Tramas RTX Enlace bajada	FER Enlace bajada (%)
1	13,41	0,67	20954	11347	54,15	10077	940	9,32
2	10,21	1,75	23292	13817	59,32	8338	95	1,13
3	11,12	1,14	18871	9972	52,84	9183	207	2,25
4	17,39	0,12	17864	8820	49,37	9138	680	7,44
5	12,7	1,68	16502	8070	48,9	9656	439	4,54

Media	12,966	1,072	19497	10405	52,916	9278,4	472,2	4,936
Varianza	6,1686	0,3804	5710700	4136200	14,2484	338700	95146	9,4693

El throughput medio de la transmisión es 12,966 Mbps y el PER inferior al 2% siendo un valor muy bueno teniendo en cuenta que ahora ambos enlaces tanto el de subida como el de bajada están alejados del router.

Es con esta tabla y más adelante representado en las gráficas las diferencias entre los mecanismos de transmisión entre el adaptador Wi-Fi USB y el Router, ya que, aunque tanto el enlace de subida como el enlace de bajada recorren la misma distancia, el FER es mucho mayor en el enlace de subida (TX → Router) mientras que en el del enlace de bajada es inferior al 10% (FER enlace de subida = inferior al 60%). Esto se debe principalmente a los mecanismos de transmisión que emplea cada dispositivo.

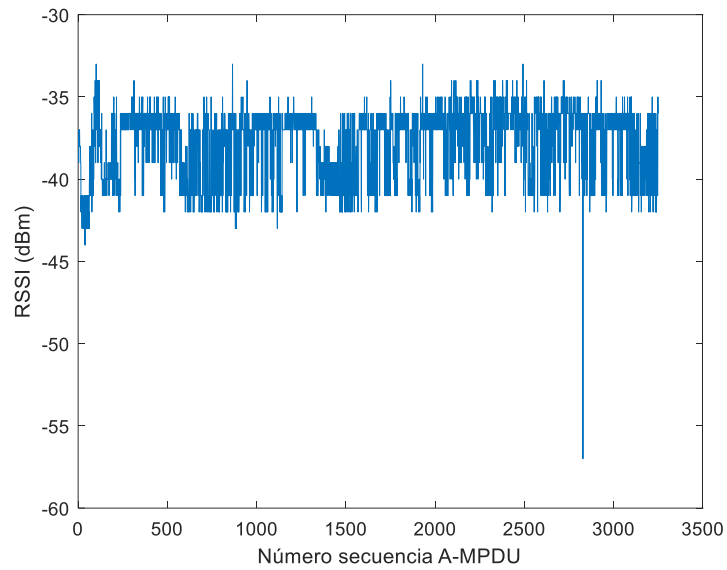
En el caso del adaptador Wi-Fi USB empleará el mecanismo de transmisión de Atheros empleando el driver ath9k\_htc. Como se verá adelante, este mecanismo de transmisión siempre intentará enviar a la mejor velocidad posible según el canal. Esto produce que haya muchas pérdidas de tramas, que se compensa con un alto número de retransmisiones como se podrá ver en la figura 6.20. Una característica del mecanismo de transmisión de Atheros es que solo emplea una velocidad idéntica, es decir, (Ver Tabla 2.3) aunque dos MCS Index tengan la misma velocidad de transmisión, solo se empleará uno, como por ejemplo entre el MCS Index 5 y 11 (velocidad de transmisión 52 Mbps). [3]

En el caso del Router, el mecanismo de transmisión que emplea es del fabricante. Este mecanismo de transmisión en vez de transmitir a la mejor velocidad posible envía las tramas a velocidades menores mejorando el FER evitando un gran número de retransmisiones. En este caso a diferencia del mecanismo de transmisión de Atheros, sí podrá emplear dos velocidades de transmisión idénticas con distinto MCS Index.

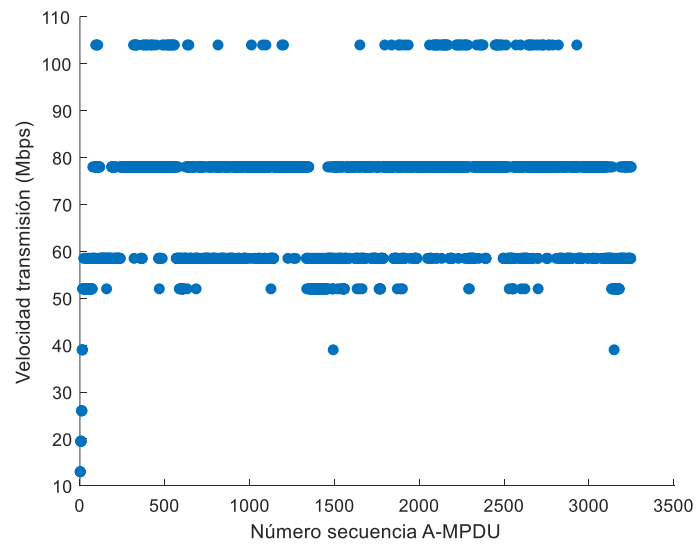
Ambos mecanismos de transmisión retransmiten la primera vez siempre a la misma velocidad de la trama original y a partir de ahí si se producen más retransmisiones de una misma trama van reduciendo la velocidad de transmisión hasta que se logre transmitir o se dé por perdida la trama. [4]

Una vez explicado esto, ahora se cogerá una medida, en este caso la medida 2, y se representará información relacionada con esta y explicará tanto para el enlace de subida como de bajada.

➤ Enlace subida

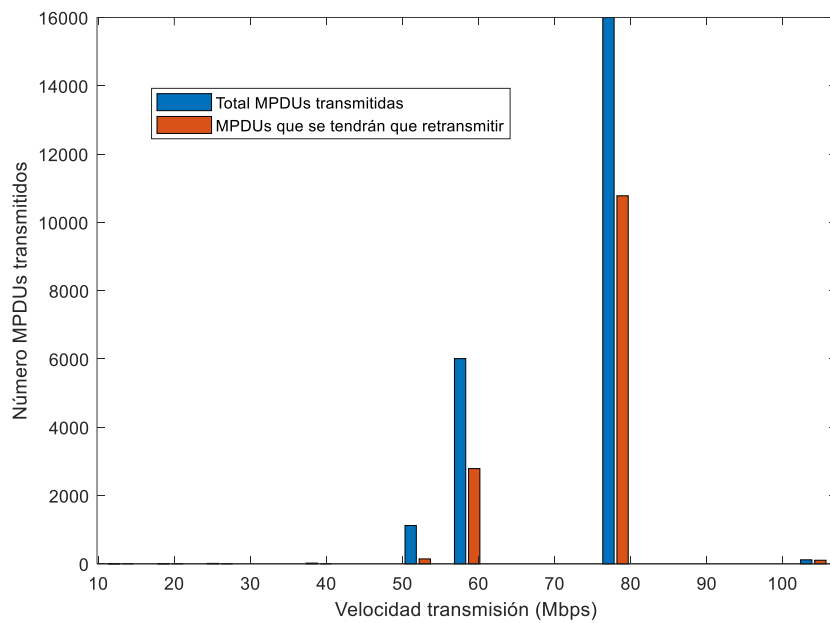


*Figura 6.16. Potencia de señal recibida en el transmisor en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura*

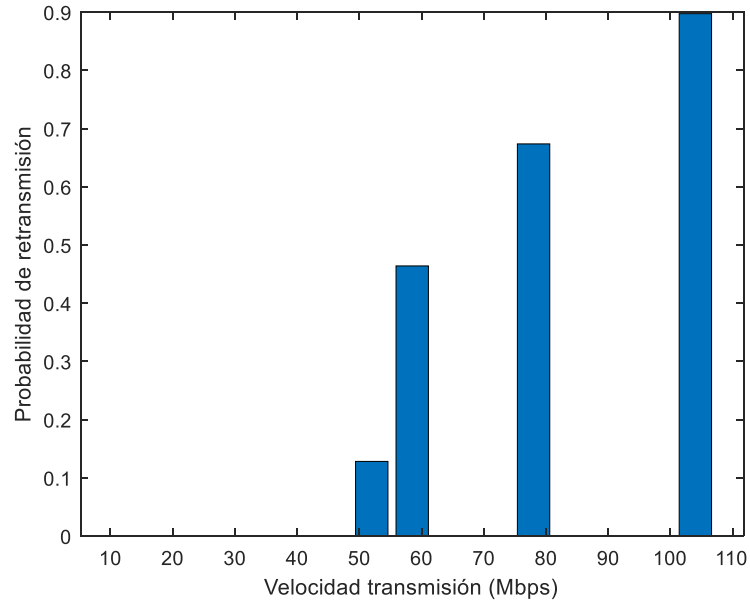


*Figura 6.17. Velocidad de transmisión en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura*

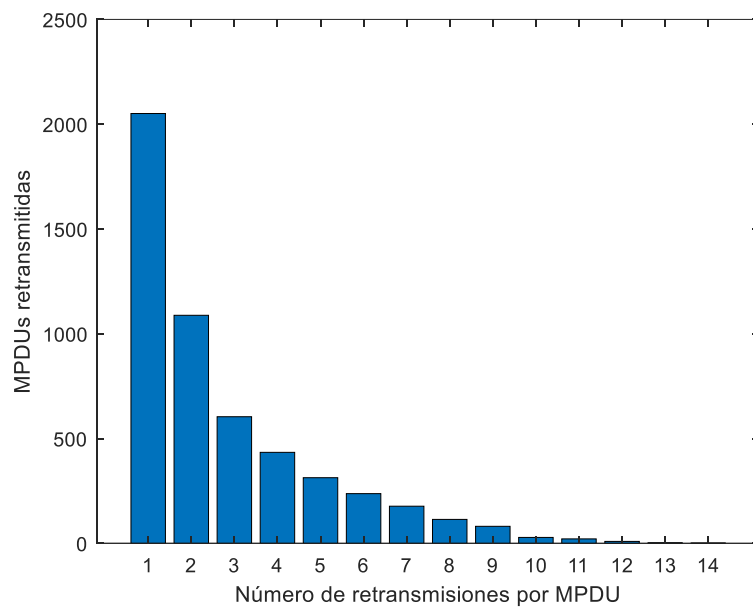




*Figura 6.18. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura*

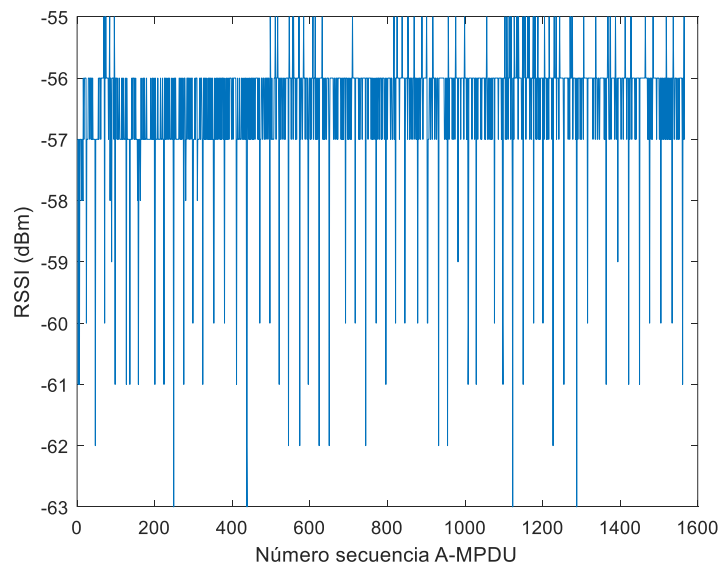


*Figura 6.19. Probabilidad de retransmisión a una determinada velocidad en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura*



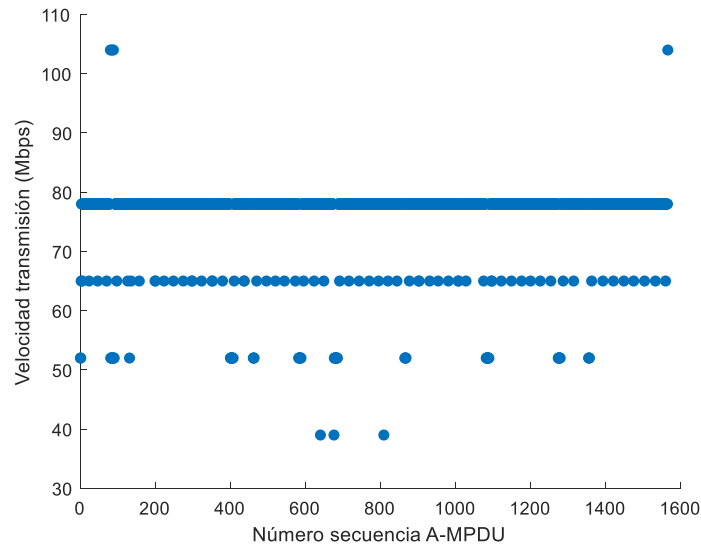
*Figura 6.20. Número de veces que se transmite una retransmisión en el escenario 2, enlace de subida, en una red Wi-Fi en modo infraestructura*

➤ Enlace bajada

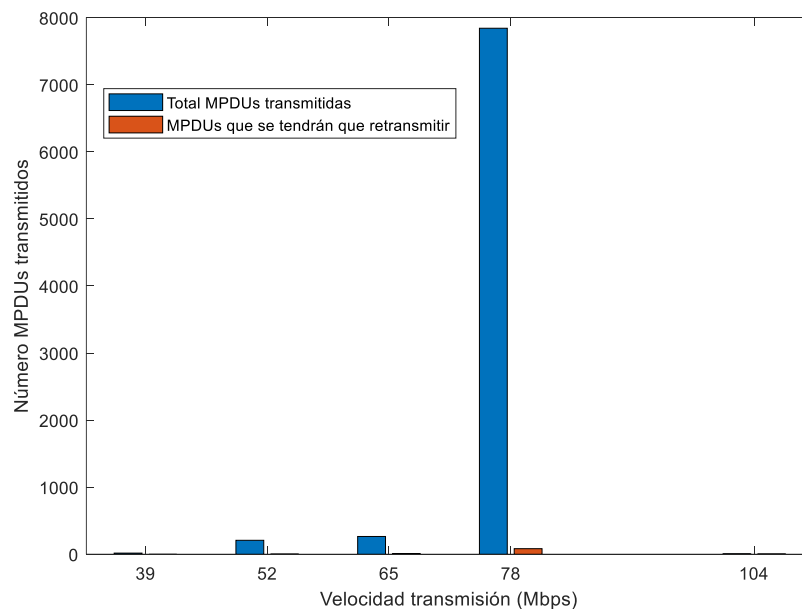


*Figura 6.21. Potencia de señal recibida en el transmisor en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura*

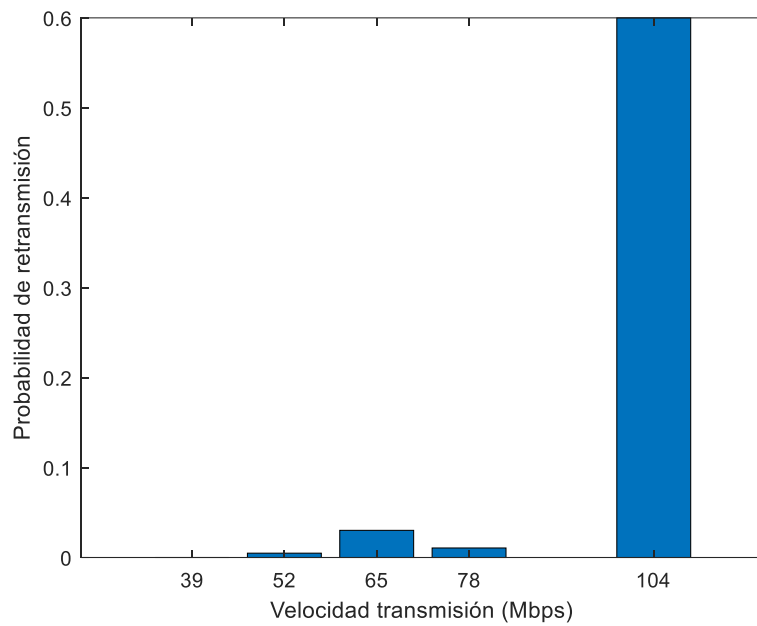
La figura 6.21 a pesar de haber sido captada en el transmisor, representa la potencia de señal recibida en el receptor. Esto es debido a que tanto el receptor y el transmisor están juntos y por ello el modo monitor capta la transferencia de paquetes con la misma potencia de señal que el receptor. Al ser la distancia en el enlace de bajada igual en el escenario 1 y como el 2 también esta figura representará la potencia de señal recibida en el receptor. Para los escenarios 3 y 4 ocurrirá igual, pero con distintos valores a los del escenario 1 y 2 debido al cambio de horario.



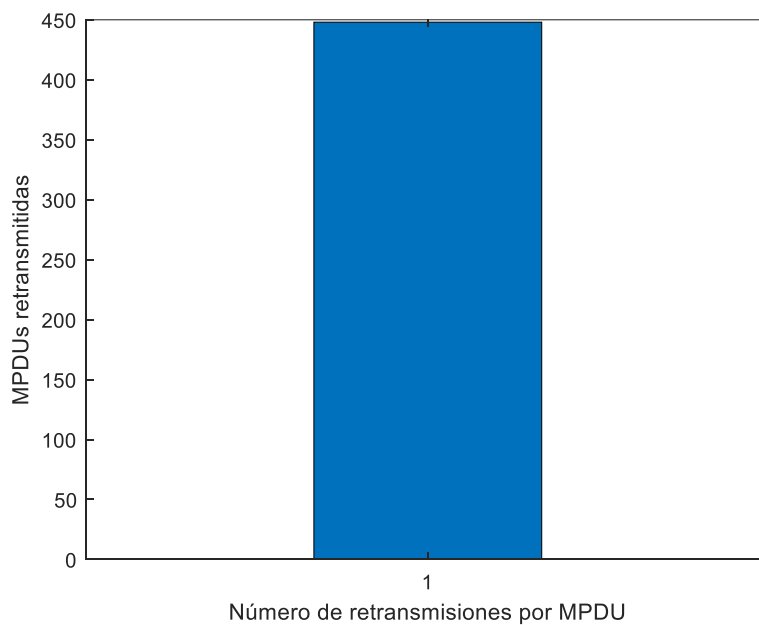
*Figura 6.22. Velocidad de transmisión en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.23. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.24. Probabilidad de retransmisión a una determinada velocidad en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.25. Número de veces que se transmite una retransmisión en el escenario 2, enlace de bajada, en una red Wi-Fi en modo infraestructura*

### 6.1.3 Análisis medidas escenario 3

Este escenario es idéntico al escenario 1 pero por la tarde. Los resultados son similares, habiendo un empeoramiento en este escenario del FER y del Throughput pero manteniéndose el PER. El FER del enlace de bajada comparado con el FER del escenario es ligeramente superior como cabría esperar, mientras que una mayor diferencia se produce en el FER del enlace de subida, pasando de un 6% hasta aproximadamente un 20%. Esto es debido al aumento de tráfico de otras redes Wi-Fi que se produce. En cuanto al Throughput solo se reduce en 5 Mbps aproximadamente.

Para este escenario se emplearán las gráficas obtenidas de la medida 2.

Tabla 6.3. Escenario 3: Transmisor y router juntos por la tarde alejados del receptor en una red Wi-Fi en modo infraestructura

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas Enlace subida	Tramas RTX Enlace subida	FER Enlace subida (%)	Tramas procesadas Enlace bajada	Tramas RTX Enlace bajada	FER Enlace bajada (%)
1	27,08	0	11569	2099	18,14	12010	2169	18,05
2	20,51	0	11702	3169	27,08	10423	633	6,07
3	16,02	0	16956	7144	42,13	10426	569	5,45
4	23,55	0,27	11563	2099	18,15	10737	1096	10,2
5	19,43	0	10755	1019	9,47	10242	515	5,02

Media	21,318	0,054	12509	3106	22,994	10768	996,4	8,958
Varianza	14,0937	0,0117	5056418	4538616	122,5612	411240	386280	24,0723

➤ Enlace subida

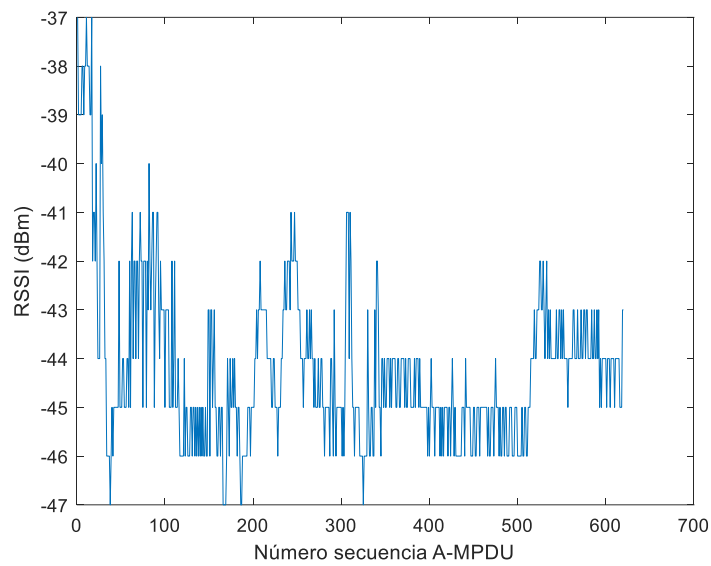


Figura 6.26. Potencia de señal recibida en el transmisor en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura

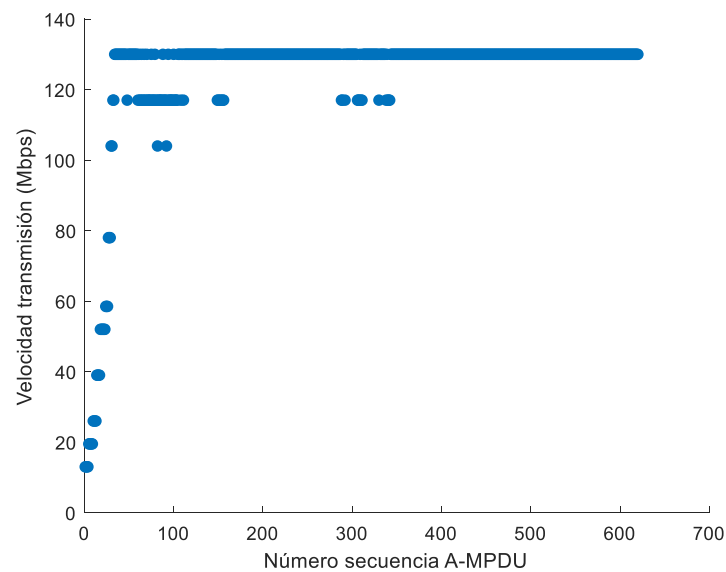
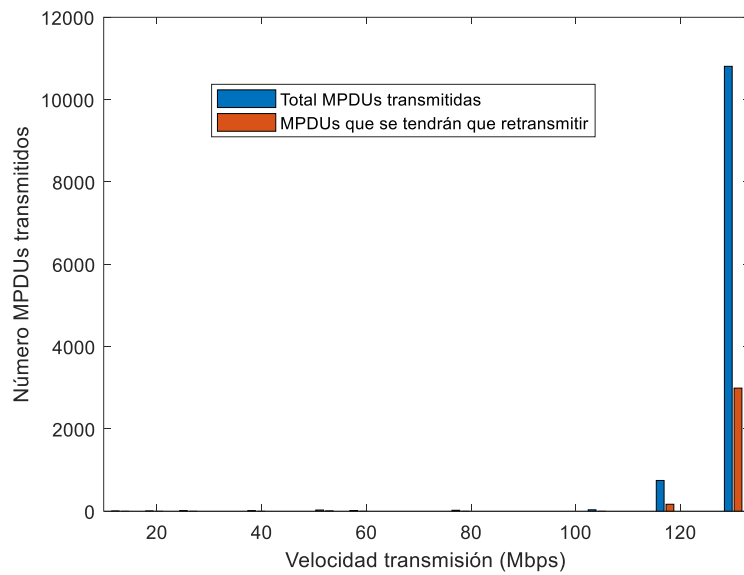
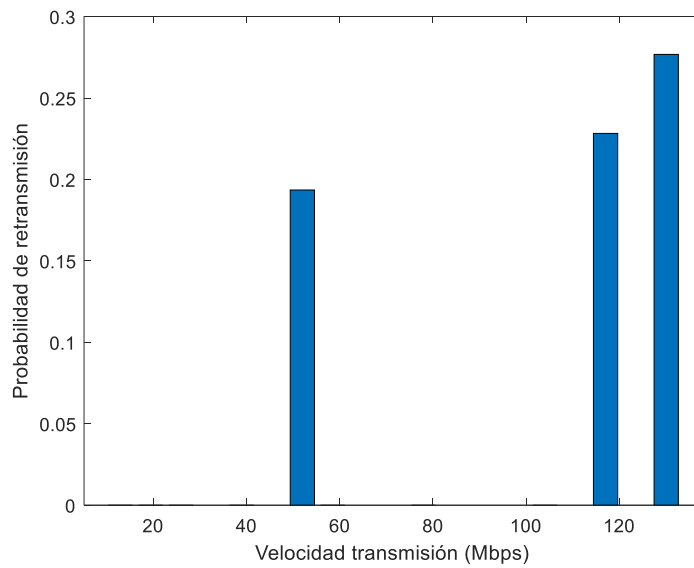


Figura 6.27. Velocidad de transmisión en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura



*Figura 6.28. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura*



*Figura 6.29. Probabilidad de retransmisión a una determinada velocidad en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura*



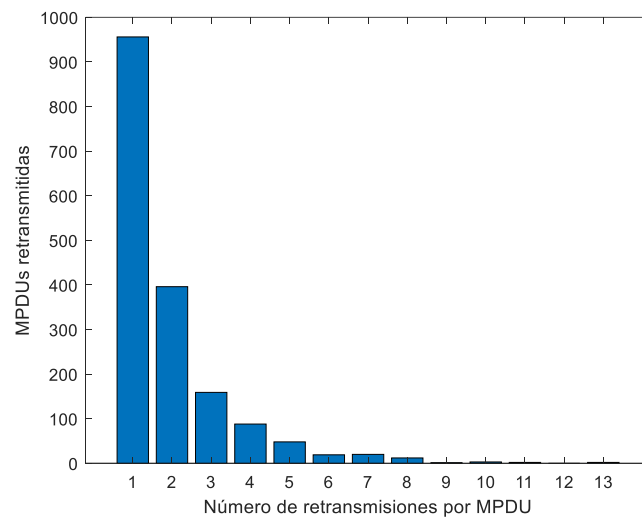


Figura 6.30. Número de veces que se transmite una retransmisión en el escenario 3, enlace de subida, en una red Wi-Fi en modo infraestructura

➤ Enlace bajada

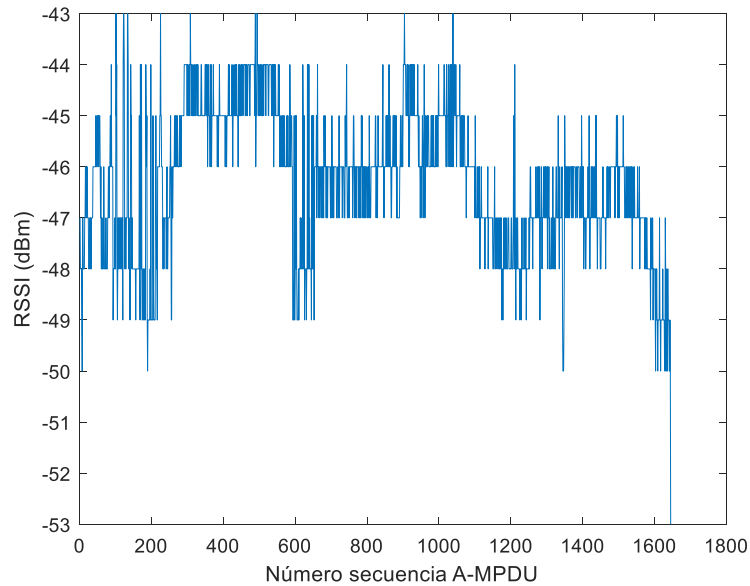


Figura 6.31. Potencia de señal recibida en el transmisor en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura

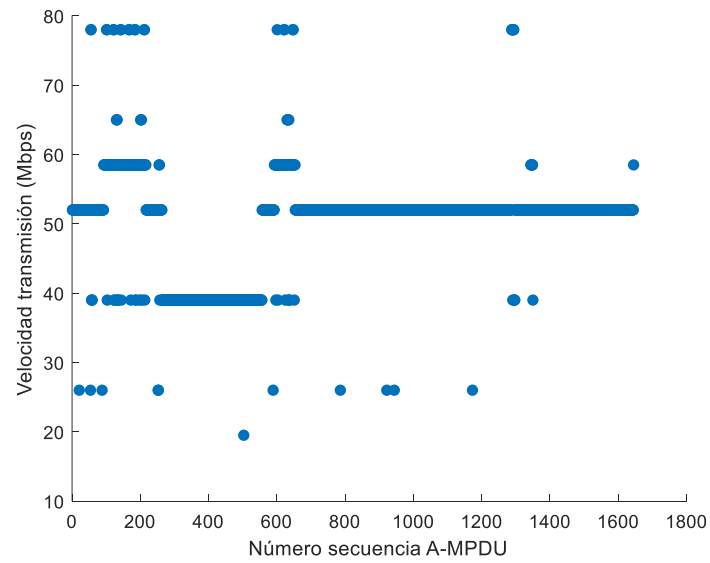


Figura 6.32. Velocidad de transmisión en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura

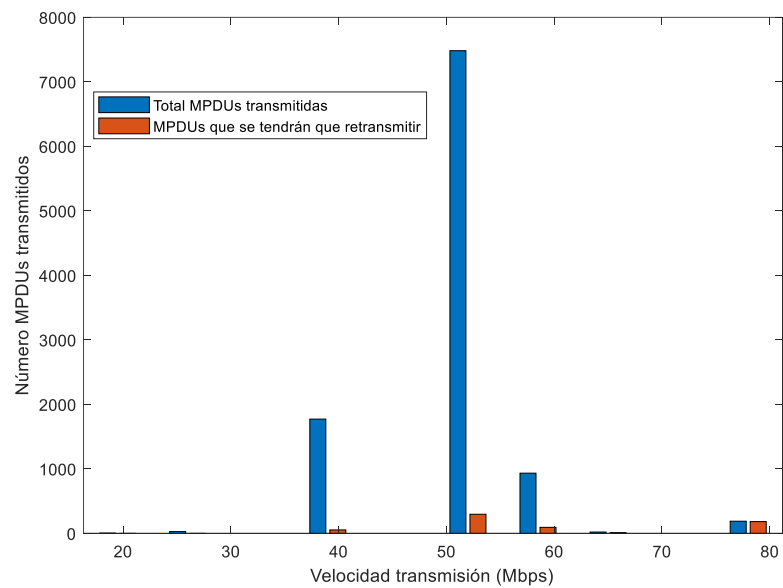
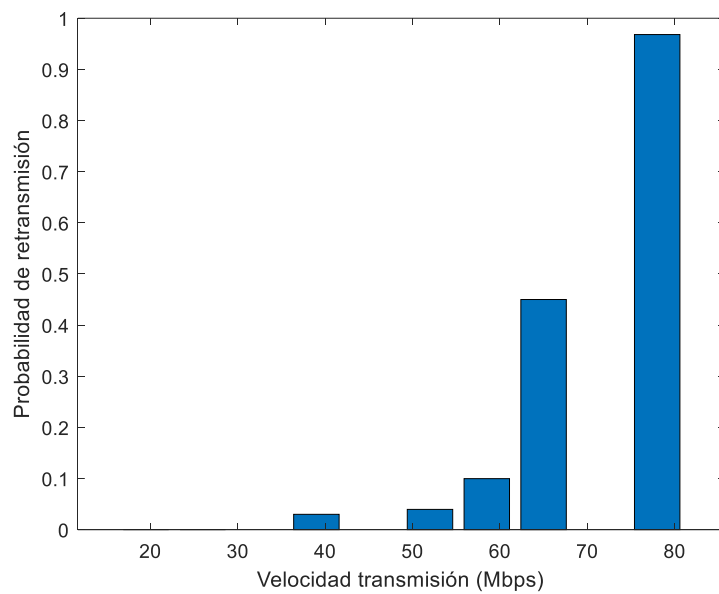
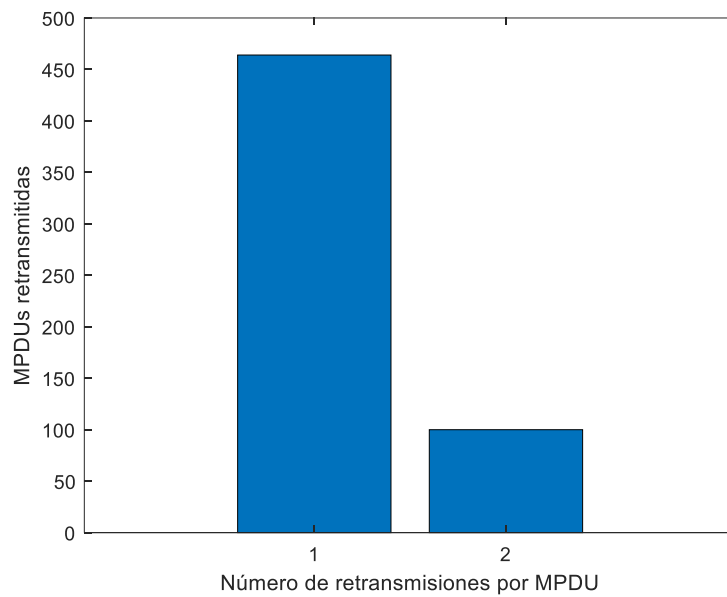


Figura 6.33. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura



*Figura 6.34. Probabilidad de retransmisión a una determinada velocidad en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.35. Número de veces que se transmite una retransmisión en el escenario 3, enlace de bajada, en una red Wi-Fi en modo infraestructura*

#### **6.1.4 Análisis medidas escenario 4**

Este escenario es idéntico al escenario 2 pero las medidas son realizadas por la tarde con una mayor presencia de tráfico. Como es obvio, los datos obtenidos son peores comparados con los del escenario 3 donde se realizan también las medidas por la tarde, pero el transmisor se encuentra al lado del router. Si se compara con el escenario 2 del cual es una copia cambiando únicamente el horario se podrá observar que el FER del enlace de subida es similar (diferencia de 1%), mientras que en el FER del enlace de bajada en el escenario 4 es peor con una diferencia del 3% aproximadamente. En este escenario también el Throughput es menor pero no por una cantidad considerable, solamente 2 Mbps.

Para este escenario se emplearán las gráficas obtenidas de la medida 2.

Tabla 6.4. Escenario 4: Transmisor y Receptor juntos por la tarde alejados del router en una red Wi-Fi en modo infraestructura

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas Enlace subida	Tramas RTX Enlace subida	FER Enlace subida (%)	Tramas procesadas Enlace bajada	Tramas RTX Enlace bajada	FER Enlace bajada (%)
1	11,32	0,74	21821	12134	55,6	10116	740	7,31
2	9,14	1,28	24693	14925	60,44	9677	543	5,61
3	10,75	0,58	20886	11004	52,68	10340	871	8,42
4	12,20	0,90	17582	7770	44,19	10530	1186	11,26
5	12,01	0,03	18408	8555	46,47	9951	670	6,73

Media	11,084	0,706	20678	10878	51,876	10123	802	7,866
Varianza	1,2099	0,1682	6441600	6605500	35,2312	88257	48113	3,7031

➤ Enlace subida

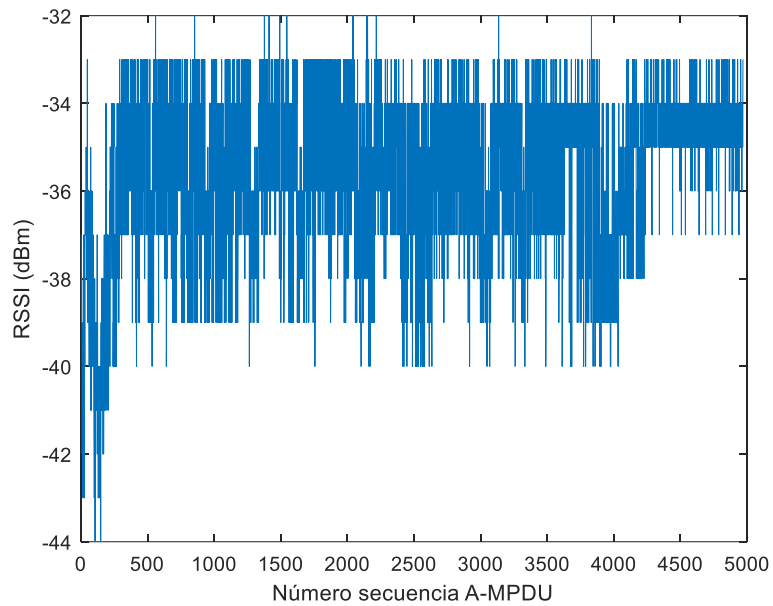


Figura 6.36. Potencia de señal recibida en el transmisor en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura

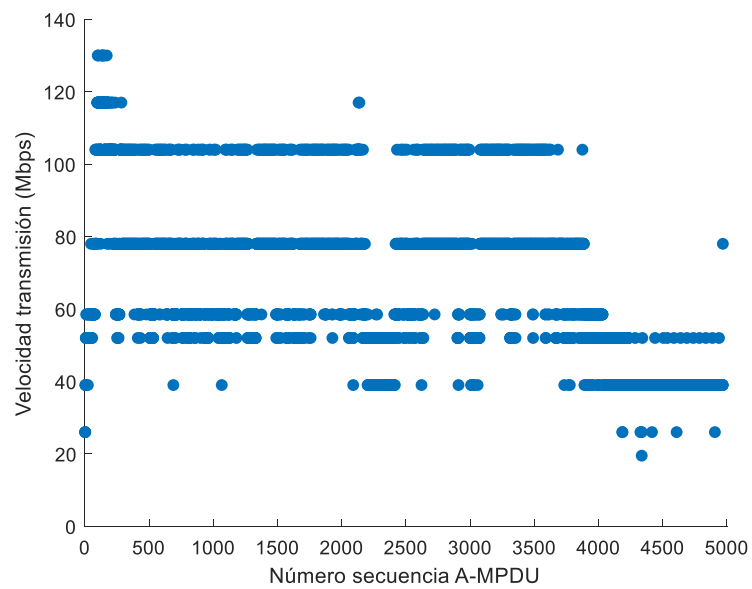
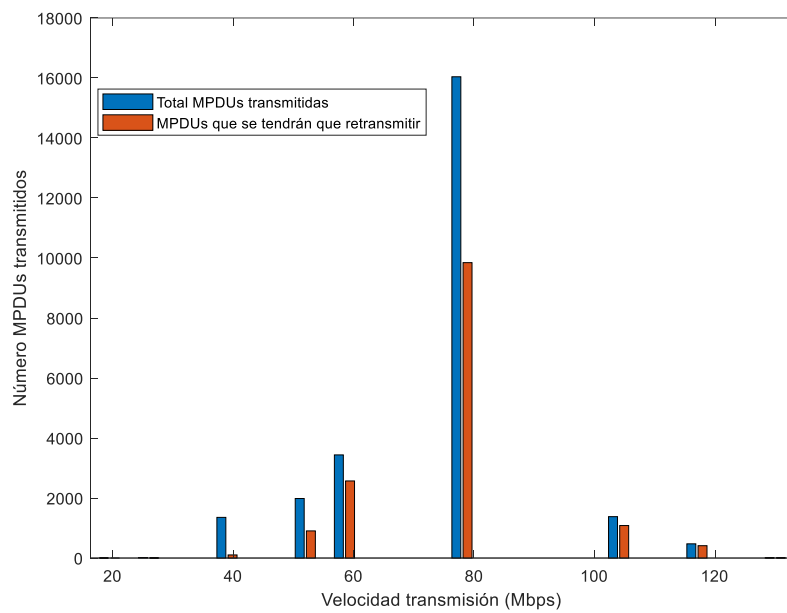
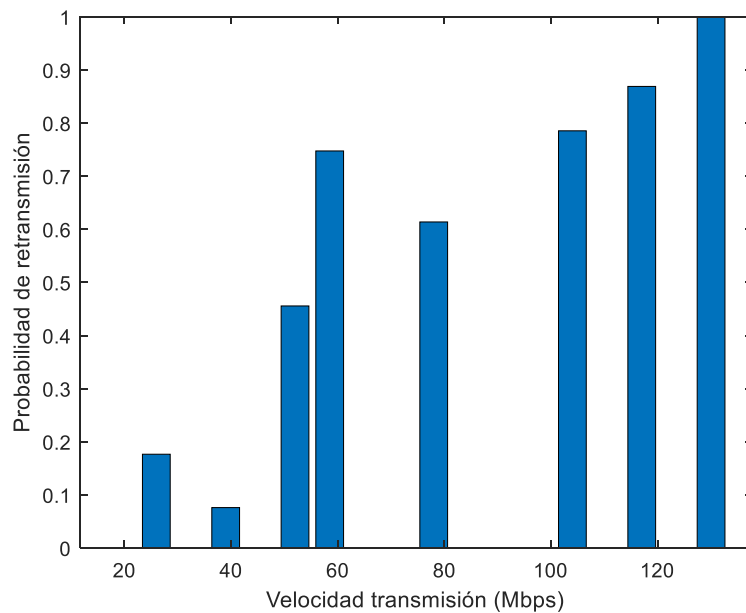


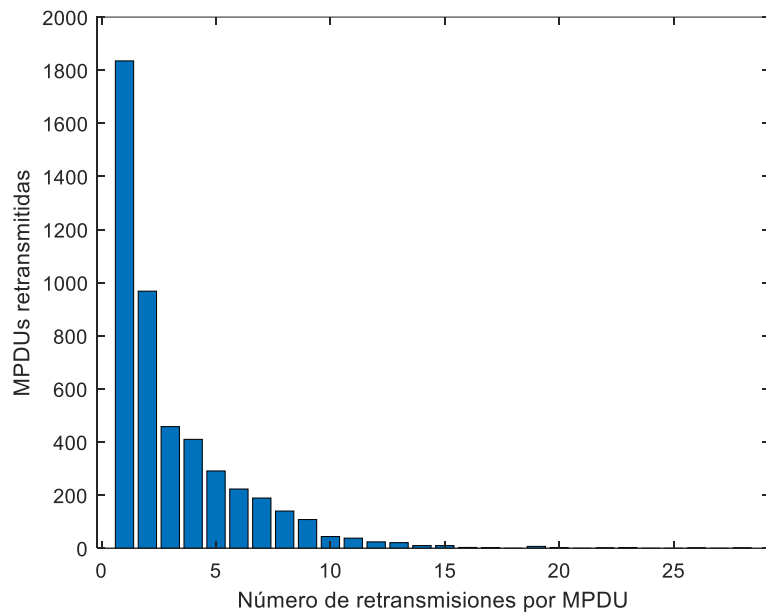
Figura 6.37. Velocidad de transmisión en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura



*Figura 6.38. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura*

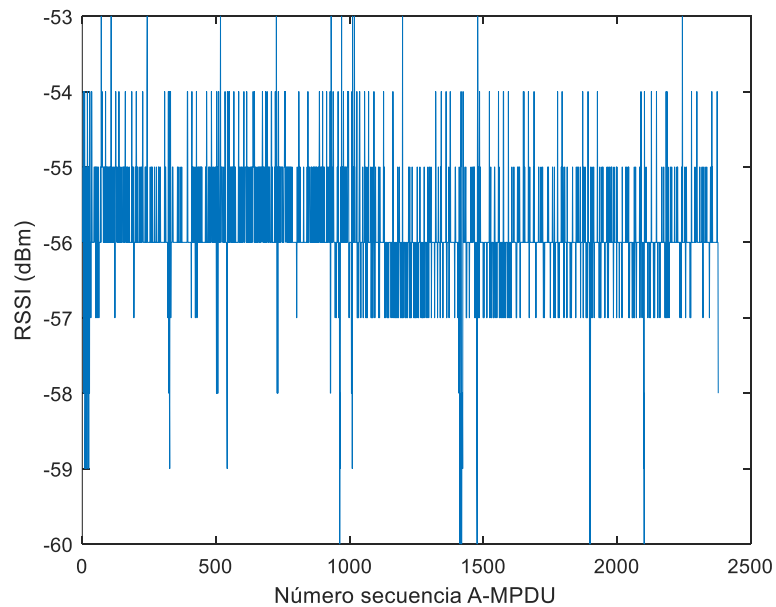


*Figura 6.39. Probabilidad de retransmisión a una determinada velocidad en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura*



*Figura 6.40. Número de veces que se transmite una retransmisión en el escenario 4, enlace de subida, en una red Wi-Fi en modo infraestructura*

➤ Enlace bajada



*Figura 6.41. Potencia de señal recibida en el transmisor en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura*



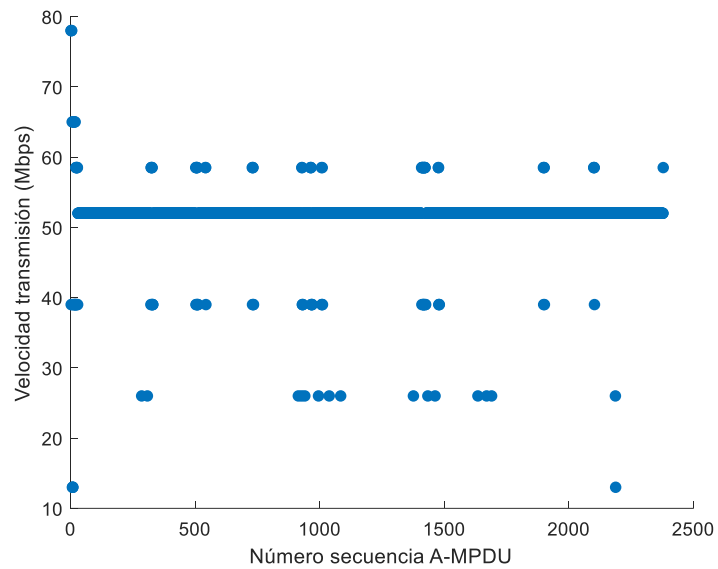


Figura 6.42. Velocidad de transmisión en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura

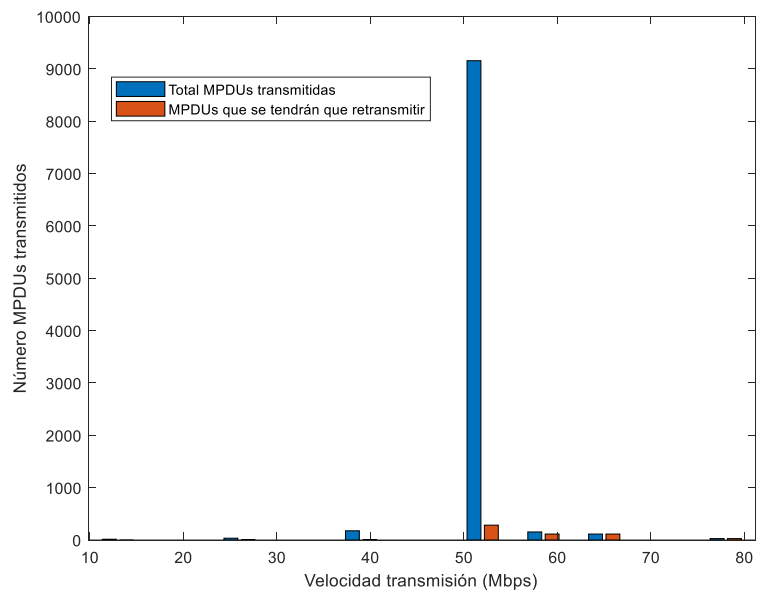
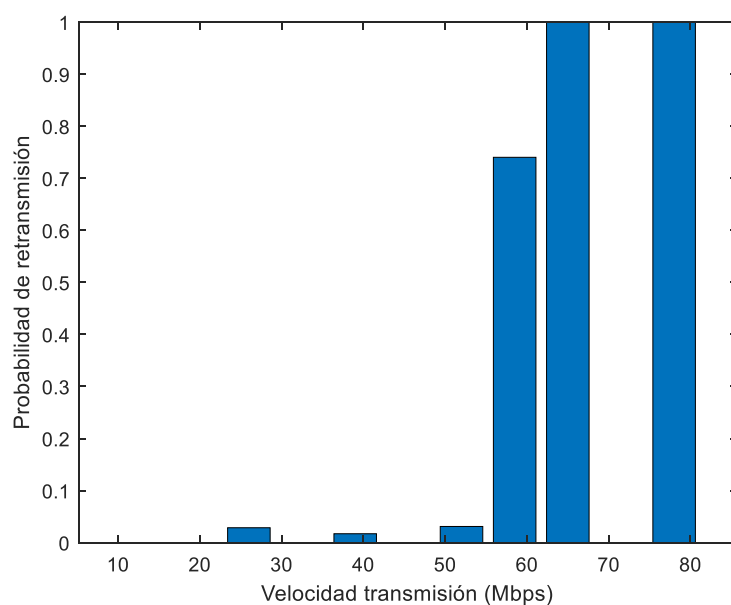
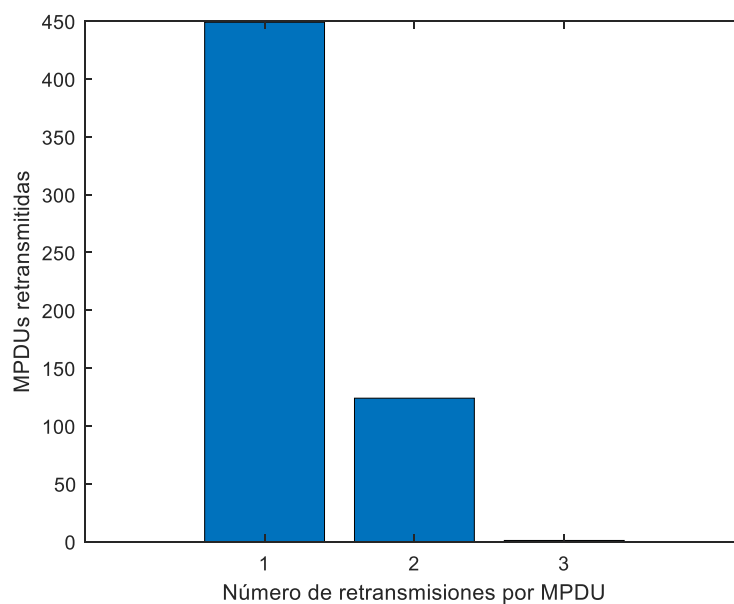


Figura 6.43. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura



*Figura 6.44. Probabilidad de retransmisión a una determinada velocidad en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura*



*Figura 6.45. Número de veces que se transmite una retransmisión en el escenario 4, enlace de bajada, en una red Wi-Fi en modo infraestructura*

### 6.1.5 Resultados Iperf en modo infraestructura

En este apartado se analizarán una serie de medidas, 5 en concreto para los escenarios 1 y 4, empleando la herramienta de medidas Iperf, con la cual se podrá obtener el dato “Jitter”, que es la variación de la latencia. Iperf, a diferencia de Nttcp, no puedes indicar cuantos paquetes exactos deseas enviar, a Iperf se le indica cuanto tiempo dura la transmisión de datos e Iperf intentará enviar el máximo posible de paquetes en ese periodo de tiempo. Las medidas realizadas en Iperf se realizarán en el escenario 1 y en el escenario 4. Se han elegido estos dos escenarios porque representan el mejor caso (escenario 1, transmisor al lado del router y receptor alejado por la mañana) y el peor caso (escenario 4, transmisor y receptor alejados del router por la tarde). Los resultados se representarán con una tabla:

Escenario 1:

*Tabla 6.5. Medidas Iperf realizadas en el escenario 1 con un tiempo de 3 segundos*

# Test	Throughput (Mbps)	Jitter (ms)	Paquetes enviados	Paquetes perdidos	PER (%)
1	40,5	1,037	10393	0	0
2	42,9	1,199	11033	0	0
3	41,7	1,265	10680	0	0
4	39,9	1,466	10270	0	0
5	40,8	1,06	10498	0	0

Media	41,16	1,2054	10575	0	0
Varianza	1,0944	0,0242	70573	0	0

Cada una de estas medidas Iperf, se ha realizado en el escenario 1, en un periodo de tiempo de tres segundos. Los datos que se representan son los paquetes enviados, paquetes perdidos, PER, Throughput y Jitter. Todos estos datos son devueltos por la herramienta de medidas Iperf por lo que no hay que hacer ningún cálculo. El Jitter medio en este caso es de 1,2 ms.

Escenario 4:

*Tabla 6.6. Medidas Iperf realizadas en el escenario 4 con un tiempo de 8 segundos*

# Test	Throughput (Mbps)	Jitter (ms)	Paquetes enviados	Paquetes perdidos	PER (%)
1	10,1	3,355	6963	29	0,42
2	12	2,061	8252	39	0,47
3	11,9	3,158	8295	31	0,37
4	13	3,489	8899	33	0,37
5	11,4	2,498	7952	26	0,33

Media	11,68	2,9122	8072,2	31,6	0,392
Varianza	0,8936	0,2971	402070	19,04	0,002

Las medidas realizadas con Iperf en el escenario 4 fueron realizadas en un periodo de 8 segundos. Aquí se observa que a pesar de enviar menos paquetes que en el escenario 1, ha necesitado un periodo de tiempo mayor para poder enviar esa inferior cantidad de paquetes, habiendo incluso pérdidas de paquetes. Esto provoca un aumento del Jitter medio a 2,9 ms (la medida 4 incluso tiene un jitter de 3,5 ms) y una reducción considerable del throughput sobre todo si es comparado con el escenario 1.

## **6.2 Medidas en modo Ad-hoc**

En este apartado se analizarán las medidas realizadas en una red Wi-Fi en modo ad-hoc, es decir, sin necesidad de ningún punto de acceso para la interconexión entre los ordenadores. Para ello habrá cuatro subapartados, cada uno correspondiente al escenario empleado según la posición del transmisor y el horario en el que se realizó las medidas.

En la realización de este conjunto de medidas se empleará una red Wi-Fi en modo Ad-hoc, que como se mencionó en la Introducción en el punto 1.1 Presentación y objetivos, consiste en una red Wi-Fi en la cual los dispositivos de la red Wi-Fi están interconectados sin necesidad de un punto de acceso (Router). En este apartado se detallarán las medidas y explicación de los resultados obtenidos en red Wi-Fi en modo Ad-hoc en un entorno doméstico.

En este caso, el beacon de la red Wi-Fi Ad-hoc variará del beacon de la red Wi-Fi en modo infraestructura por el nombre de la red (SSID=tfg) y compartirá las mismas características excepto las relacionadas con 802.11, que contendrá la información mostrada anteriormente en el Probe Request (Ver 6.1.2), ya que se recuerda que la red Wi-Fi ad-hoc es creada por el adaptador Wi-Fi USB y por lo tanto empleará las características de este.

### **6.2.1 Análisis medidas escenario 1**

En este apartado se analizarán 5 medidas, como siempre enviando 10000 paquetes UDP con Nttcp, realizadas en el escenario 1 (receptor y transmisor juntos por la mañana) en una red en modo Ad-hoc. Estas medidas solo tendrán un flujo de datos, ya que no es necesario un punto de acceso (router) evitando que haya un enlace de subida y un enlace de bajada como ocurría en una red Wi-Fi en modo infraestructura. Esto reducirá el tamaño de la tabla de resultados, pero los resultados que se mostrarán serán los mismos (Ver 6.1.1 para explicación tabla resultados y obtención de datos).

Este escenario será el mejor de todos teniendo en cuenta incluso los escenarios de la red Wi-Fi en modo infraestructura, por lo que los resultados que se mostrarán serán el mejor caso posible gracias a la cercanía entre el transmisor y el receptor y no depender de un punto de acceso, además de facilitar la lectura de los resultados debido a que solo se emplea el mecanismo de transmisión de Atheros.

Tabla 6.7. Resultados de las medidas en el escenario 1 en una red Ad-hoc

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas	Tramas RTX	FER (%)
1	78,07	0	10353	392	3,78
2	77,07	0	10449	450	4,3
3	82,01	0	10307	303	2,93
4	72,81	0	10197	365	3,57
5	81,79	0	10328	388	3,75

Media	78,35	0	10327	379,6	3,666
Varianza	11,52	0	6572,2	2252,2	0,1946

Como se puede observar, los resultados extraídos son excelentes. No hay ninguna pérdida de paquete, un throughput altísimo superando en 50 Mbps al throughput medio del mejor caso (escenario 1) de las medidas de la red en modo infraestructura y una FER inferior al 4%, que teniendo en cuenta como se verá en las gráficas que representarán la medida 1 al final de este apartado, debido al mecanismo de transmisión de Atheros, como ya se mencionó, se intentará la transmisión a la máxima velocidad posible (130 Mbps) y compensarlo con múltiples retransmisiones. En este caso la conexión es tan buena que a pesar de transmitir siempre a la máxima velocidad posible casi no hay retransmisiones.

Ahora se representarán las gráficas obtenidas a partir de la medida 1:

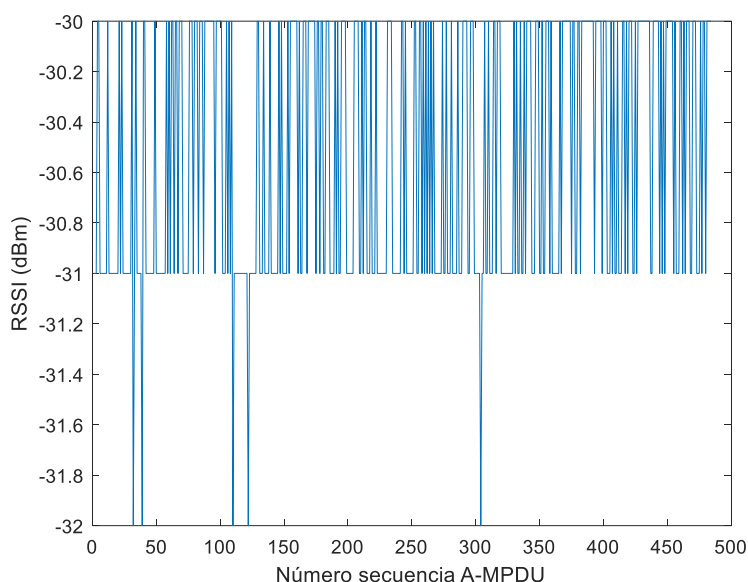


Figura 6.46. Potencia de señal recibida en el transmisor en el escenario 1 en una red Wi-Fi en modo Ad-hoc

La figura 6.46 representa tanto la potencia de señal del transmisor como la potencia de señal del receptor debido a la cercanía entre estos. Ocurrirá lo mismo en escenario 3 donde se produce el mismo escenario, pero con un horario vespertino.

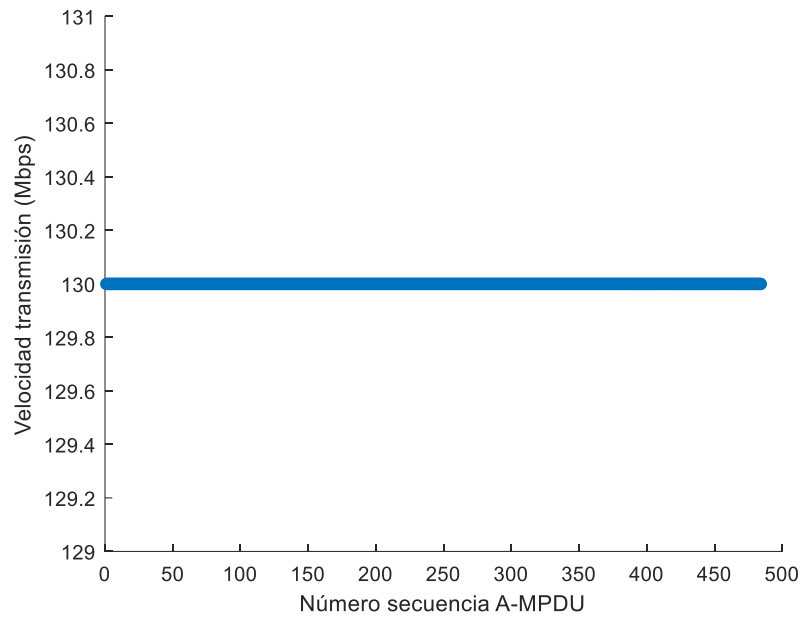


Figura 6.47. Velocidad de transmisión en el escenario 1 en una red Wi-Fi en modo Ad-hoc

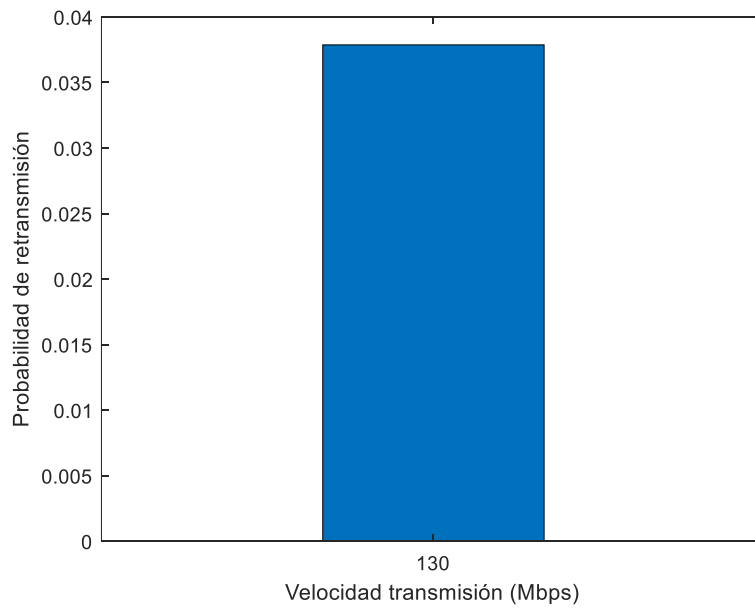
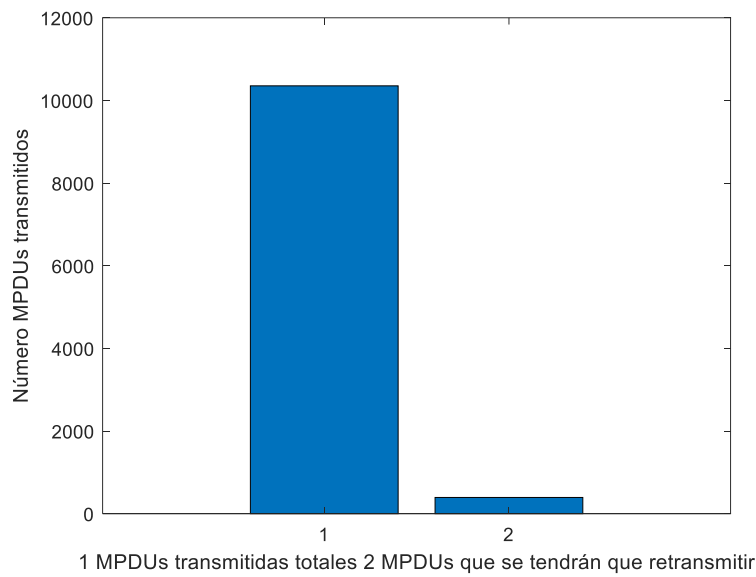
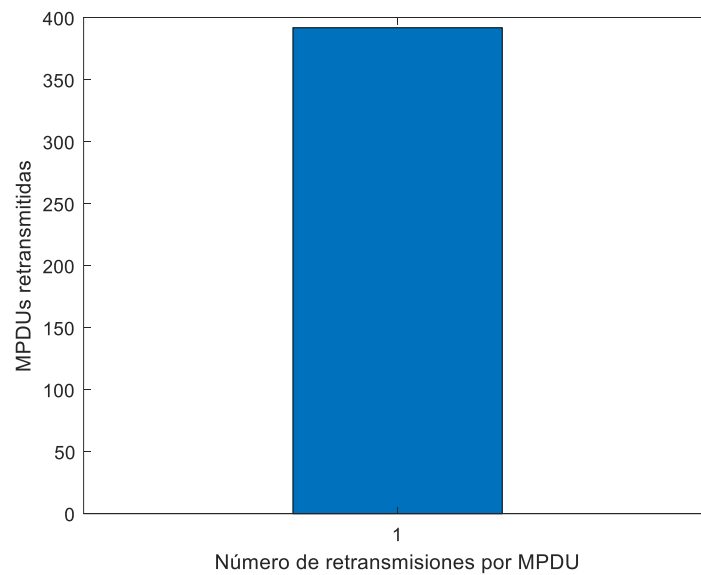


Figura 6.48. Probabilidad de retransmisión a una determinada velocidad en el escenario 1 en una red Wi-Fi en modo Ad-hoc



*Figura 6.49. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 1 en una red Wi-Fi en modo Ad-hoc*

Esta figura es una variación que sólo aparece cuando hay tramas de una única velocidad de transmisión, en este caso, a 130 Mbps.



*Figura 6.50. Número de veces que se transmite una retransmisión en el escenario 1 en una red Wi-Fi en modo Ad-hoc*

## 6.2.2 Análisis medidas escenario 2

En este escenario se realizarán otras 10000 medidas con Nttcp, con el transmisor y el receptor alejados, por lo que obviamente las medidas resultantes tendrán peores resultados que el escenario 1.

Tabla 6.8. Resultados de las medidas en el escenario 2 en una red Ad-hoc

# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas	Tramas RTX	FER (%)
1	14,39	0	18178	8431	46,38
2	15,25	0	16285	6413	39,37
3	10,81	0	19848	9905	49,9
4	9,61	0	18989	9132	48,09
5	15,34	0	17616	7792	44,23

Media	13,08	0	18183	8334,6	45,594
Varianza	5,7453	0	1469200	1419700	13,1976

Como se puede observar obviamente comparada con la tabla del escenario 1, (Tabla 6.7), donde solo varía la distancia entre receptor y transmisor, se verifica que el throughput se reduce de 70 Mbps a 13 Mbps, una reducción considerable. Se observa que a pesar de tener un FER muy alto (45%) lo que significa casi la mitad de las tramas capturadas son retransmisiones, no se ha perdido ningún paquete.

Para la representación de las gráficas se van a emplear los datos obtenidos de la medida 1:

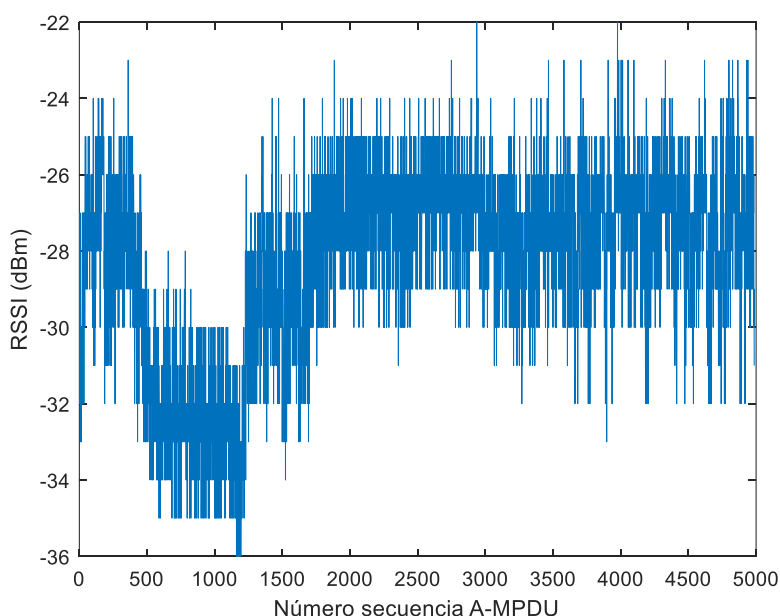
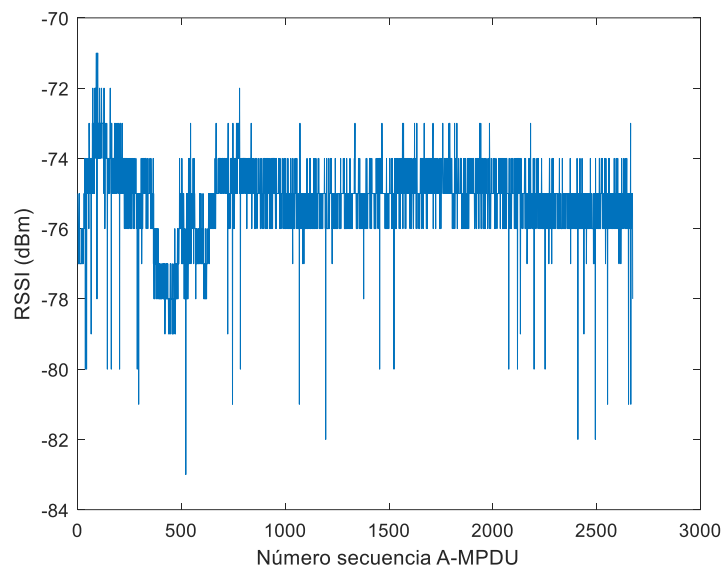


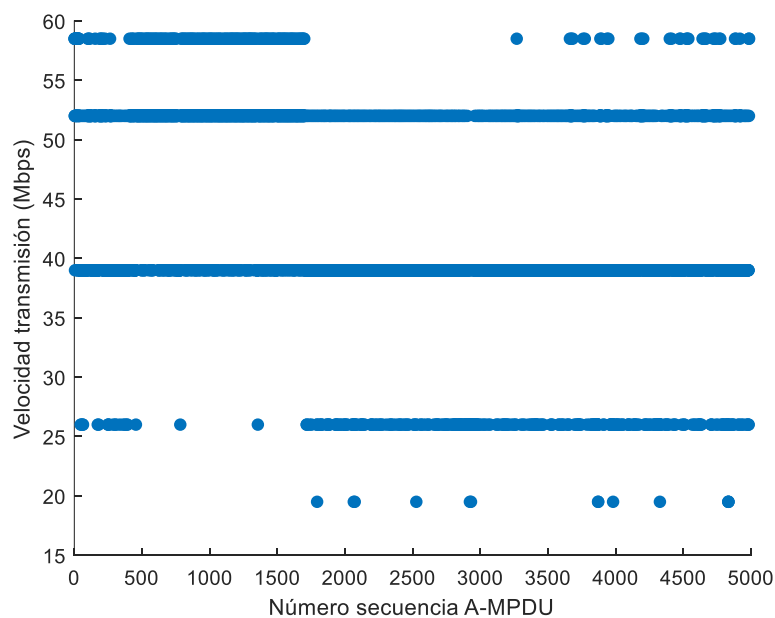
Figura 6.51. Potencia de señal recibida en el transmisor en el escenario 2 en una red Wi-Fi en modo Ad-hoc





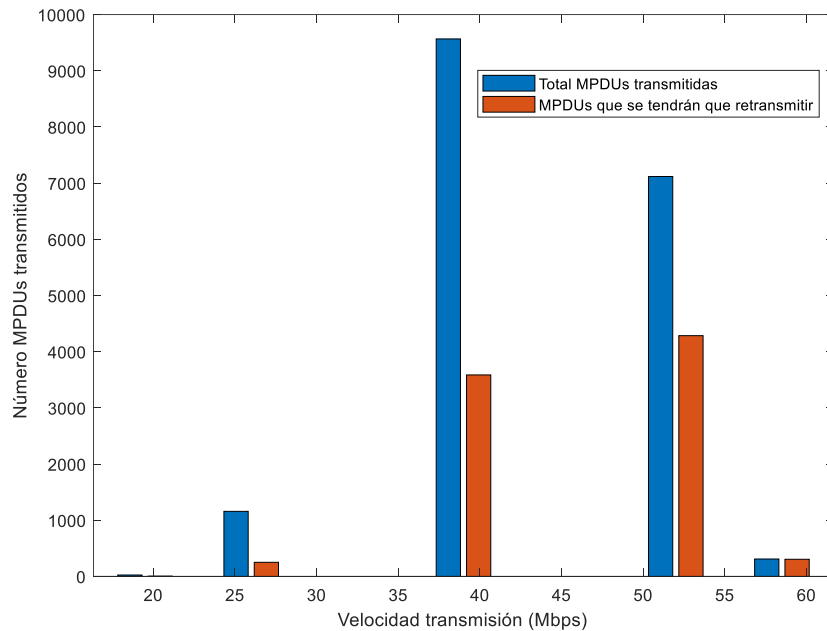
*Figura 6.52. Potencia de señal recibida en el receptor en el escenario 2 en una red Wi-Fi en modo Ad-hoc*

En este caso se ha realizado una segunda medida capturando el tráfico en el ordenador receptor para obtener la potencia de señal recibida en el receptor que corresponde con la figura 6.51. Para el escenario 4 se realizará de nuevo otra medida.

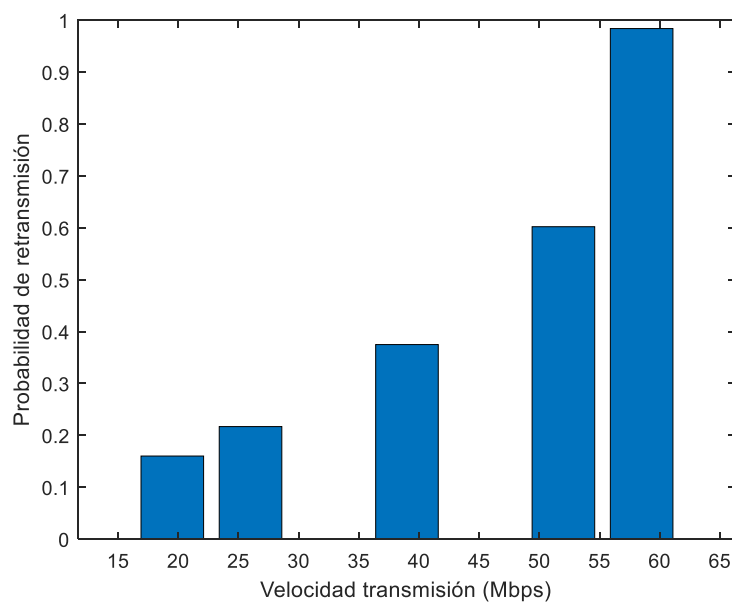


*Figura 6.53. Velocidad de transmisión en el escenario 2 en una red Wi-Fi en modo Ad-hoc*

En esta imagen se puede apreciar que hay aproximadamente 5000 A-MPDUs, frente a las 18178 tramas, lo cual significa que una A-MPDU contiene de una media unas 3,6 tramas. Esto es debido a qué cómo se explicó en el apartado 2.6.2, la reducción de la velocidad de transmisión debido al aumento de la distancia entre el receptor y transmisor, las tramas MAC necesitan más tiempo para transmitirse y, por lo tanto, la A-MPDU contendrá menos tramas.



*Figura 6.54. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 2 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.55. Probabilidad de retransmisión a una determinada velocidad en el escenario 2 en una red Wi-Fi en modo Ad-hoc*

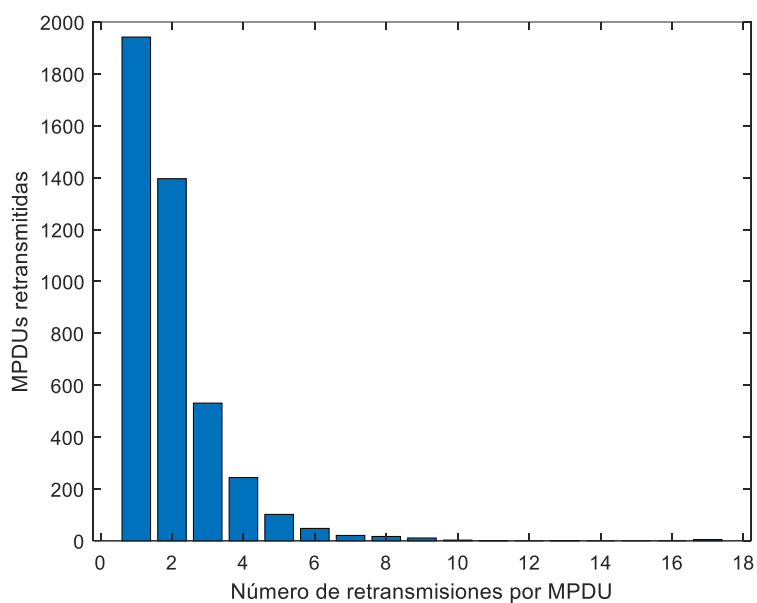


Figura 6.56. Número de veces que se transmite una retransmisión en el escenario 2 en una red Wi-Fi en modo Ad-hoc

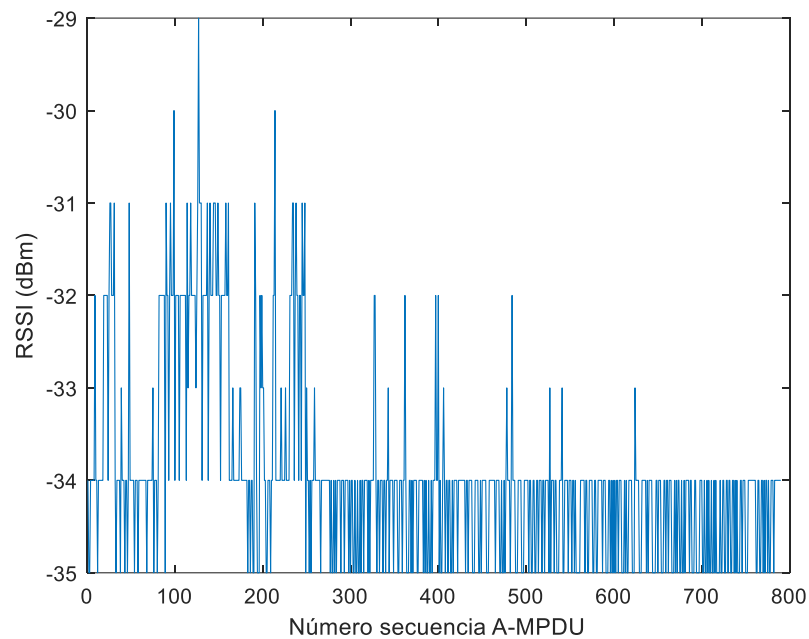
### 6.2.3 Análisis medidas escenario 3

Tabla 6.9. Resultados de las medidas en el escenario 3 en una red Ad-hoc

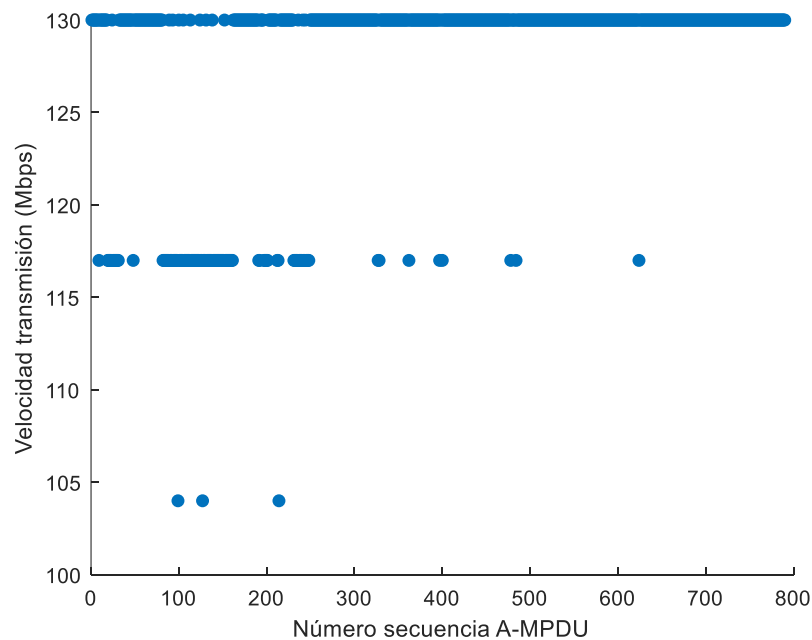
# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas	Tramas RTX	FER (%)
1	61,56	0,	11593	1695	14,62
2	67,54	0	11270	1266	11,23
3	66,26	0	11786	1794	15,22
4	41,72	0	15792	5789	36,65
5	60,73	0	12817	2813	21,94
Media	59,562	0	12652	2671,4	19,932
Varianza	86,441	0	2733600	2687600	81,9337

Este escenario es idéntico al escenario 1 con la diferencia que las medidas se han realizado por la tarde cuando hay más redes Wi-Fi activas. Como se puede observar, el Throughput en este escenario es peor en unos 20 Mbps comparado con el escenario 1 y el FER mayor llegando hasta casi el 20%. A pesar del empeoramiento de los resultados, aun así, no se ha perdido ningún paquete.

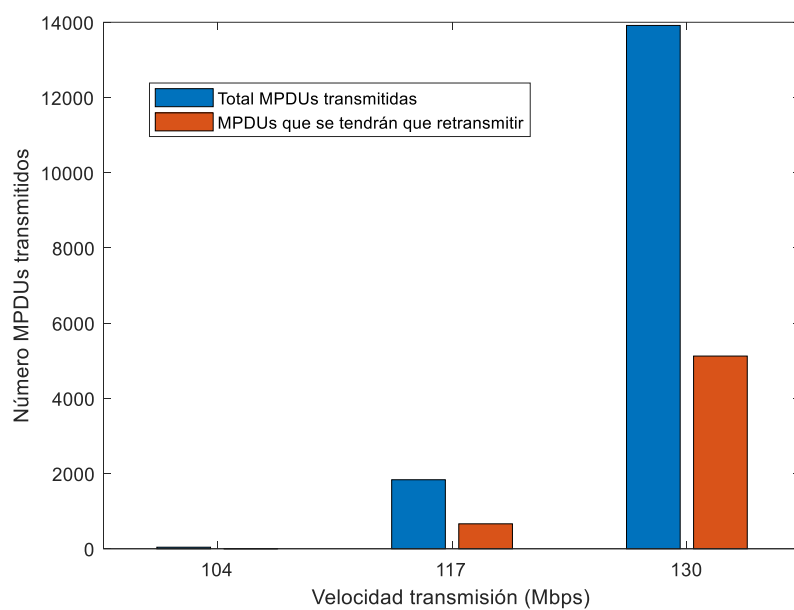
Se representarán los resultados obtenidos de la medida 4 en este escenario:



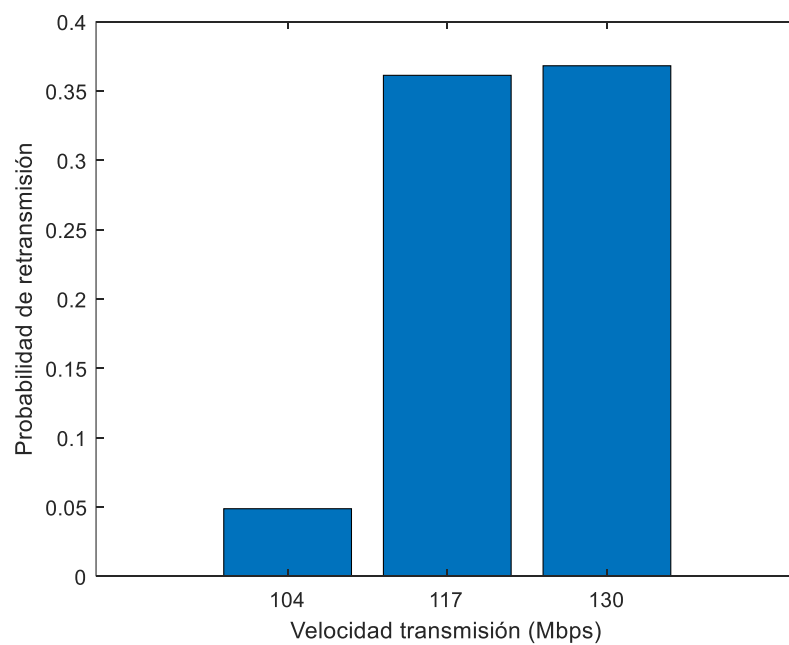
*Figura 6.57. Potencia de señal recibida en el transmisor en el escenario 3 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.58. Velocidad de transmisión en el transmisor en el escenario 3 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.59. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 3 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.60. Probabilidad de retransmisión a una determinada velocidad en el escenario 3 en una red Wi-Fi en modo Ad-hoc*

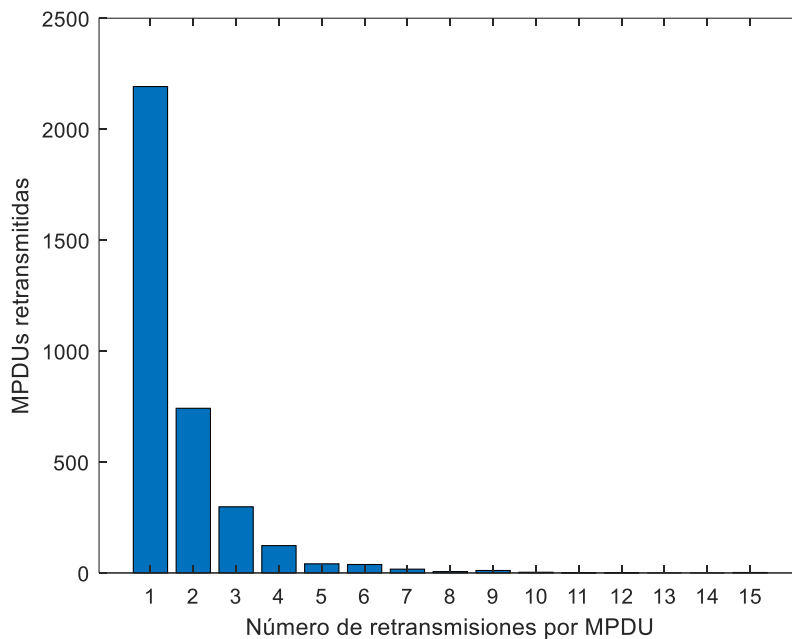


Figura 6.61. Número de veces que se transmite una retransmisión en el escenario 3 en una red Wi-Fi en modo Ad-hoc

#### 6.2.4 Análisis medidas escenario 4

Tabla 6.10. Resultados de las medidas en el escenario 4 en una red Ad-hoc

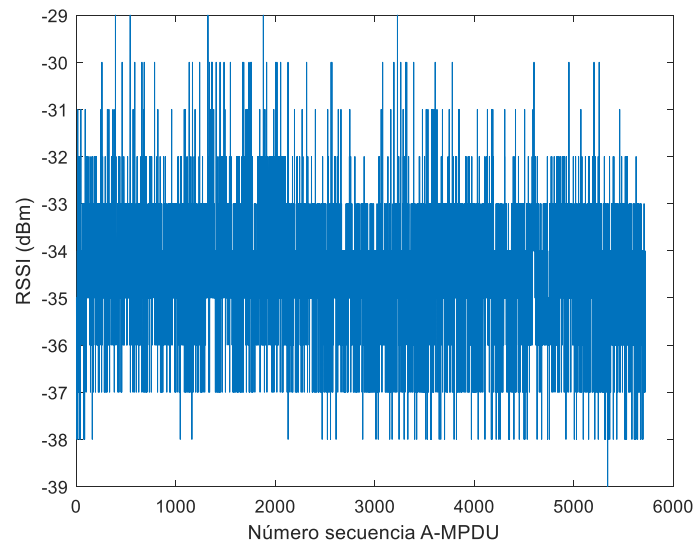
# Test	Throughput (Mbps)	Pérdida IP (%)	Tramas procesadas	Tramas RTX	FER (%)
1	15,74	0	15022	5143	34,23
2	14,11	0	16692	6776	40,59
3	13,3	0	17750	7792	43,89
4	10,23	0	19104	9123	47,75
5	10,23	0	18103	8186	45,21

Media	12,722	0	17334	7404	42,334
Varianza	4,7578	0	1931000	1844700	21,7484

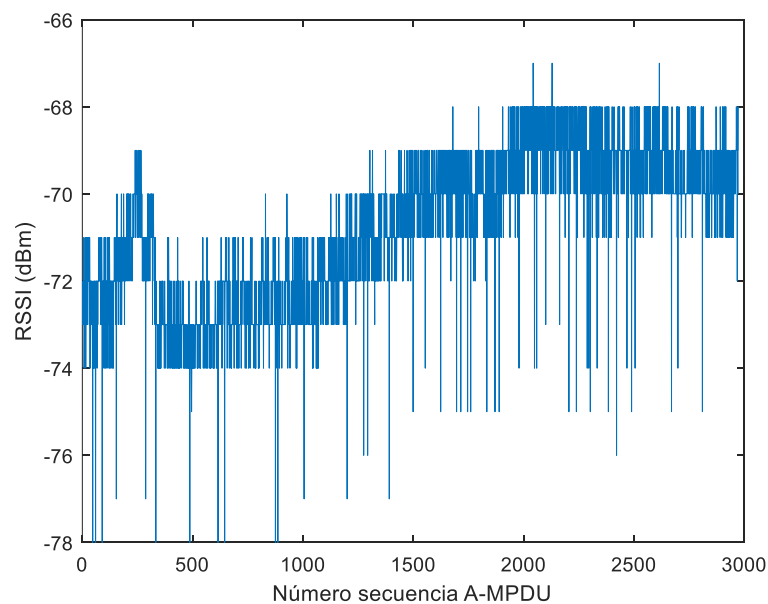
Este escenario es prácticamente idéntico al escenario 2 pero realizado por la tarde. En este caso tanto las medidas realizadas en el escenario 2 como en el escenario 4 son similares. Esto puede deberse a que la conexión ya es de tan mala calidad que realizar la medida por la tarde el aumento de redes, como ya se observó en el apartado 3.3, que no era un cambio

drástico, no influye tanto. También influye que el escenario en el que el transmisor y el receptor se encuentran alejados trabajando en una red Wi-Fi modo Ad-hoc, el transmisor se encuentre más en el interior de la casa donde llega peor la interferencia de otras redes Wi-Fi mientras que los escenarios de la red Wi-Fi en modo infraestructura es totalmente diferente. [5]

Se representarán los resultados obtenidos de la medida 3 en este escenario:



*Figura 6.62. Potencia de señal recibida en el transmisor en el escenario 4 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.63. Potencia de señal recibida en el receptor en el escenario 4 en una red Wi-Fi en modo Ad-hoc*

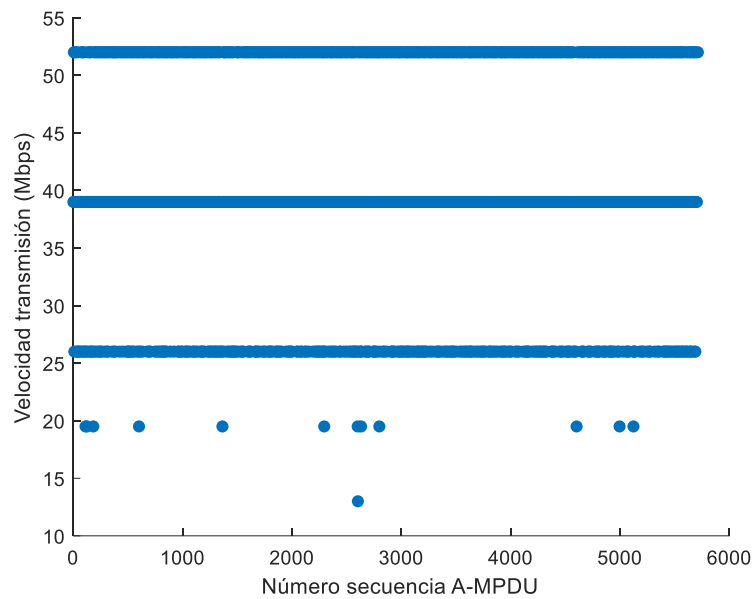


Figura 6.64. Velocidad de transmisión en el transmisor en el escenario 4 en una red Wi-Fi en modo Ad-hoc

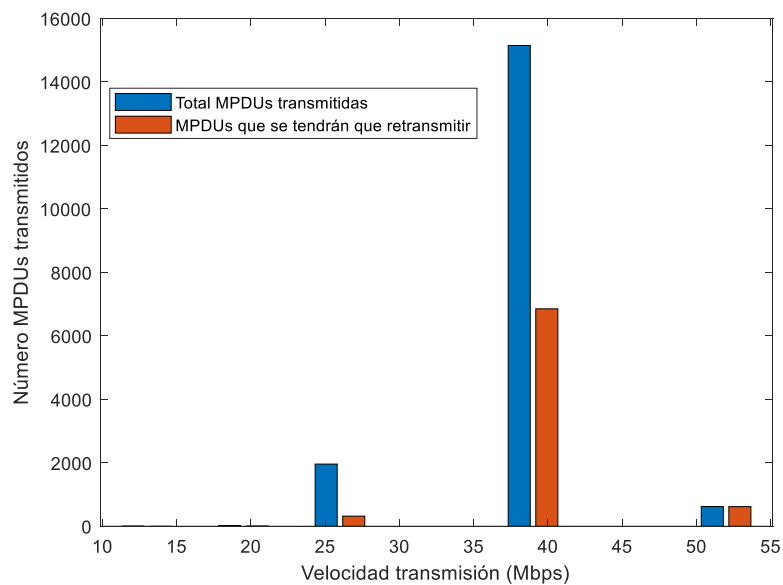
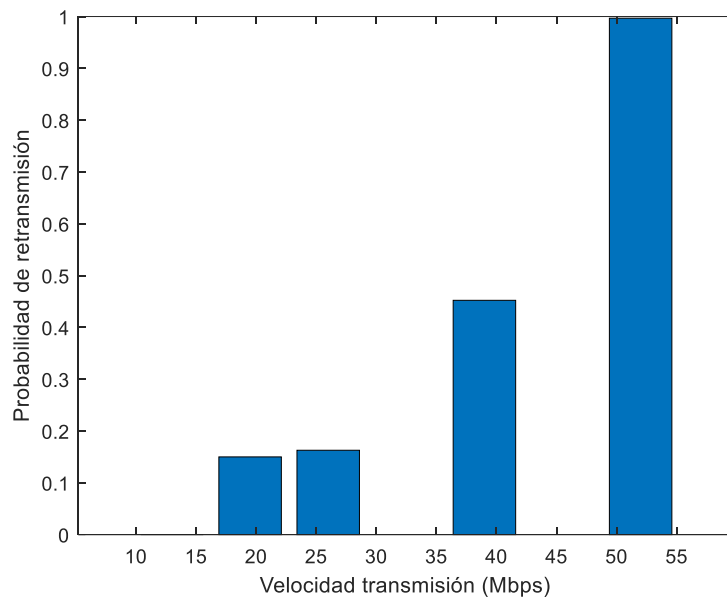
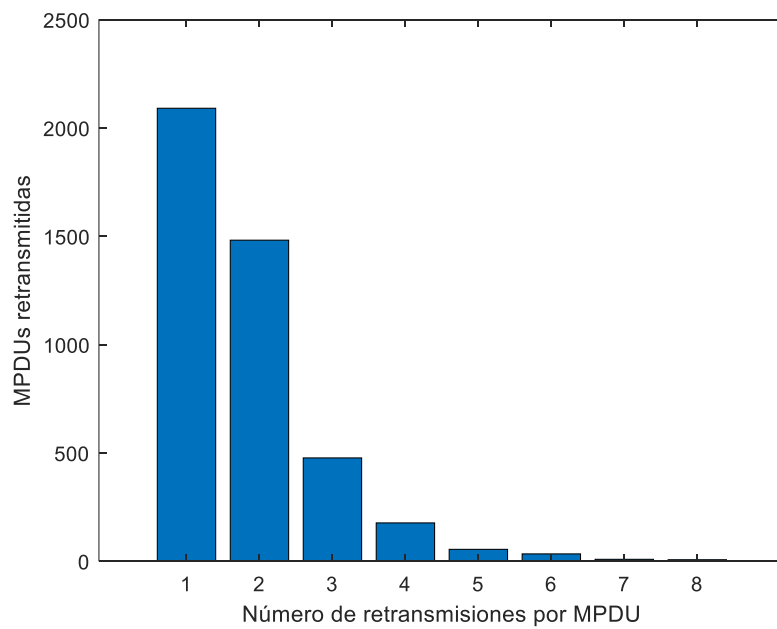


Figura 6.65. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en el escenario 4 en una red Wi-Fi en modo Ad-hoc





*Figura 6.66. Probabilidad de retransmisión a una determinada velocidad en el escenario 4 en una red Wi-Fi en modo Ad-hoc*



*Figura 6.67. Número de veces que se transmite una retransmisión en el escenario 4 en una red Wi-Fi en modo Ad-hoc*

## 6.2.5 Resultados Iperf en modo Ad-hoc

En este apartado se analizarán una serie de medidas, 5 en concreto para el escenario 1 y el escenario 4, empleando la herramienta de medidas Iperf, con la cual se podrá obtener el dato “Jitter”, que es la variación de la latencia. Las medidas realizadas en Iperf se realizarán en el escenario 1 y en el escenario 4. Se han elegido estos dos escenarios porque representan el mejor caso (escenario 1, transmisor y receptor juntos por la mañana) y el peor caso (escenario 4, transmisor y receptor alejados por la tarde, en teoría). Los resultados se representarán con una tabla:

Escenario 1:

*Tabla 6.11. Medidas Iperf realizadas en el escenario 1 con un tiempo de 3 segundos*

# Test	Throughput (Mbps)	Jitter (ms)	Paquetes enviados	Paquetes perdidos	PER (%)
1	74,2	0,623	18956	0	0
2	79,6	0,63	20336	0	0
3	77,4	0,587	19799	0	0
4	77,5	0,634	19835	0	0
5	73,4	0,807	18753	0	0

Media	76,42	0,6562	19536	0	0
Varianza	5,2576	0,006	349610	0	0

Cada una de estas medidas Iperf, se ha realizado en el escenario 1, en un periodo de tiempo de tres segundos. El Jitter medio en este caso es de 0,65 ms. Si se compara con el mejor caso de la red Wi-Fi en modo infraestructura (Ver 6.1.5), se observa una notable mejora frente al jitter medio de 1,2 ms en el mejor caso del modo infraestructura.

Escenario 4:

*Tabla 6.12. Medidas Iperf realizadas en el escenario 4 con un tiempo de 8 segundos*

# Test	Throughput (Mbps)	Jitter (ms)	Paquetes enviados	Paquetes perdidos	PER (%)
1	11,2	2,891	7647	0	0
2	16,5	3,405	11311	0	0
3	16,6	2,296	11408	0	0
4	16	2,689	10925	0	0
5	15,8	2,927	10754	0	0

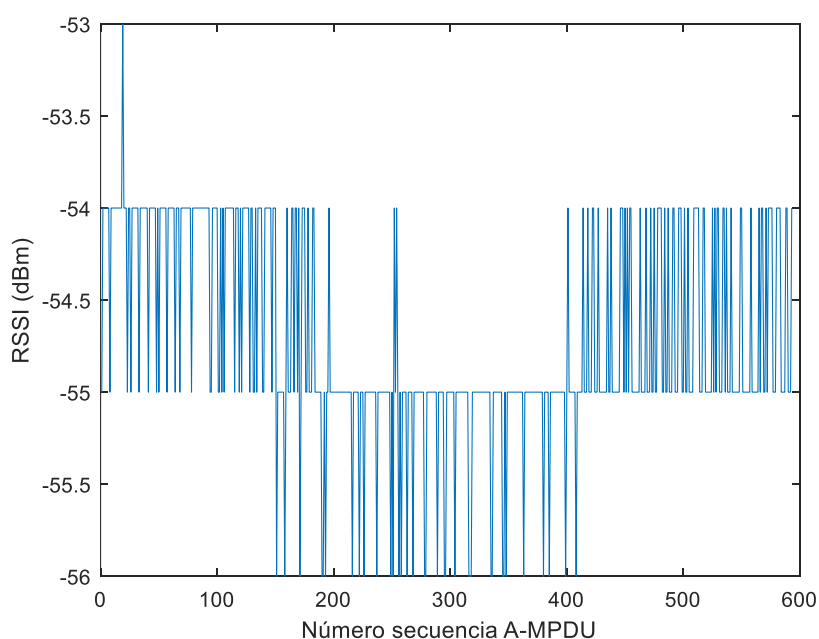
Media	15,22	2,8416	10409	0	0
Varianza	4,1296	0,1296	1965106	0	0

Cada una de estas medidas Iperf, se ha realizado en el escenario 4, receptor y transmisor alejados, en un periodo de tiempo de ocho segundos. Como se observa obviamente en la tabla 6.12 el jitter medio de 2,8 ms es peor que el de la tabla 6.11 como cabría esperar. Si se compara con su tabla homóloga de la red Wi-Fi en modo infraestructura (Ver tabla 6.4), se obtiene el dato que el Jitter medio es de 2,9 ms, es decir una diferencia de 0,1 ms, lo cual no es una gran diferencia.

## 6.2.6 Entorno ideal

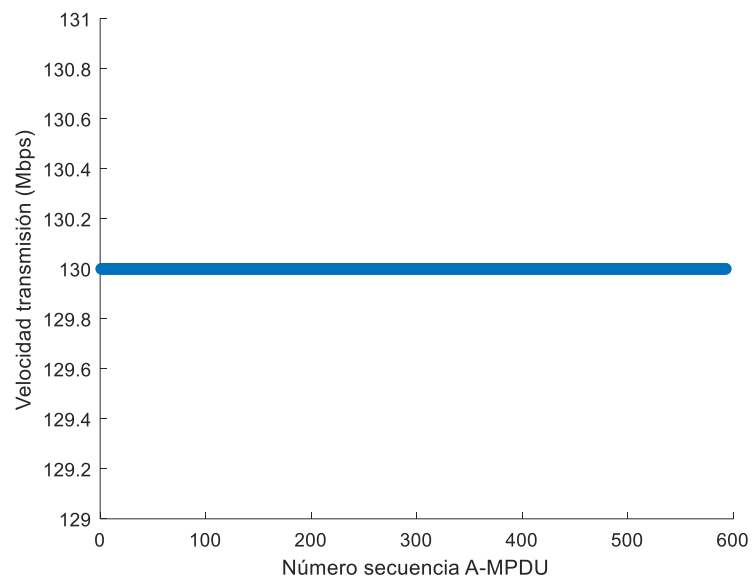
Para terminar, se ha realizado una medida en un entorno ideal sin interferencias empleando el modo Ad-hoc. Este entorno corresponderá a un laboratorio de la Universidad de Cantabria en el cual las medidas realizadas dentro de este no se verán influenciadas por ninguna red Wi-Fi exterior. El objetivo de esta medida será obtener el mejor rendimiento posible por lo que el ordenador transmisor y el ordenador receptor estarán lo más cercano posible.

La medida se ha realizado con la herramienta de medidas Iperf, debido a que esta herramienta proporciona además de la información básica, el valor del Jitter. Los resultados obtenidos con Iperf muestran un Throughput de valor 100 Mbps y un Jitter de 0,176 ms. Como cabe esperar comparando estos resultados con los obtenidos en las previas medidas, estos son los mejores resultados conseguidos. Las gráficas representadas corresponderán a los datos extraídos de la transferencia de paquetes realizada:

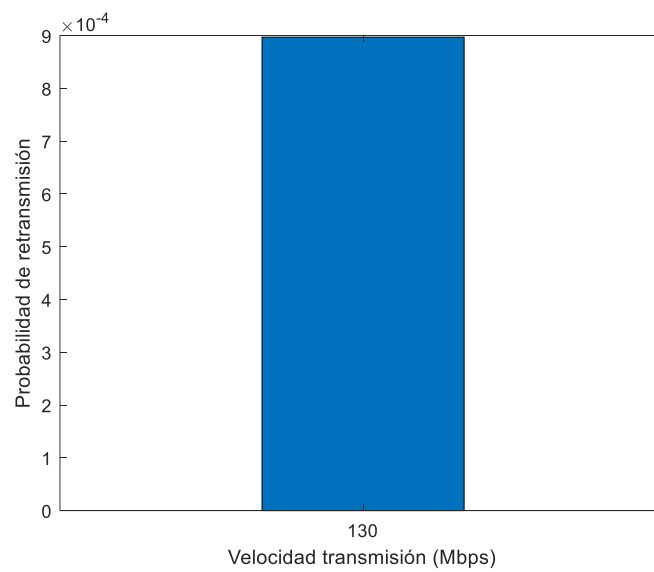


*Figura 6.68. Potencia de señal recibida en el transmisor en una red Wi-Fi en modo Ad-hoc en un entorno ideal*

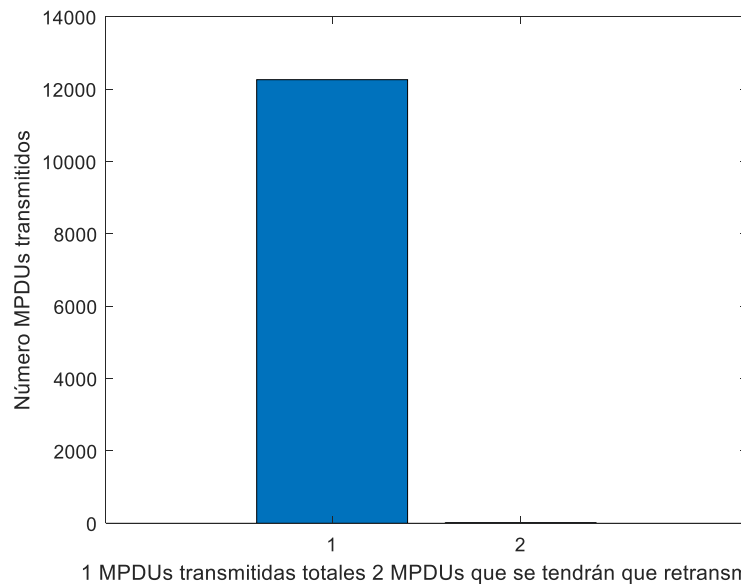
Aquí al igual que en las medidas realizadas en el modo Ad-hoc donde el receptor y el transmisor están juntos, esta potencia de señal representa tanto la potencia de transmisión y de recepción.



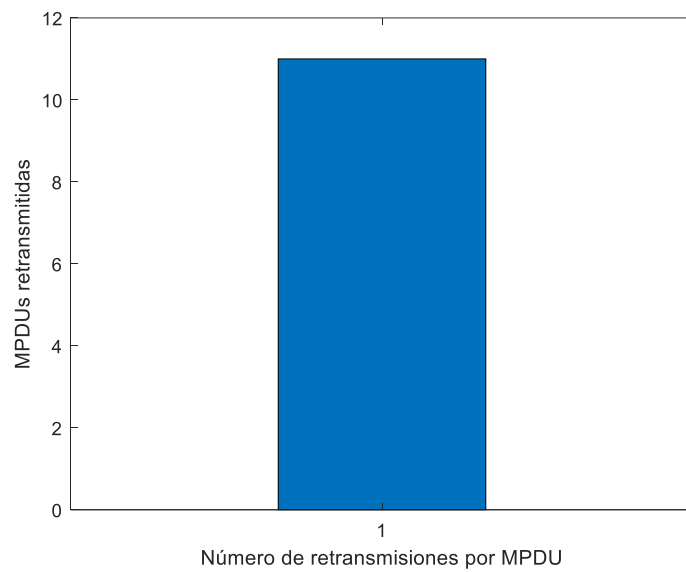
*Figura 6.69. Velocidad de transmisión en el transmisor en una red Wi-Fi en modo Ad-hoc en un entorno ideal*



*Figura 6.70. Probabilidad de retransmisión a una determinada velocidad en una red Wi-Fi en modo Ad-hoc en un entorno ideal*



**Figura 6.71. Tramas transmitidas y cantidad de ellas que se retransmitirán a una determinada velocidad en una red Wi-Fi en modo Ad-hoc en un entorno ideal**



**Figura 6.72. Número de veces que se transmite una retransmisión en una red Wi-Fi en modo Ad-hoc en un entorno ideal**

Como se puede comprobar, es una medida casi perfecta, al máximo de velocidad de transmisión posible y con simplemente 11 retransmisiones lo cual convierte los resultados en casi inmejorables.

Sabiendo el Throughput y la velocidad de transmisiones constante se podrá calcular la sobrecarga (“overhead”) presente en el mejor caso posible. Para ello se aplicará la siguiente fórmula:

$$Overhead = 1 - \frac{Throughput}{Velocidad\ de\ transmisión\ media}$$

Sabiendo que el Throughput es 100 Mbps y la velocidad de transmisión 130 Mbps, el valor de sobrecarga obtenido es de 0,23 en el mejor caso posible.

Si se compara la medida realizada en un entorno ideal con la medida mejor obtenida que en este caso corresponde a la medida 3 del escenario 1 en una red en modo Ad-hoc, se observa la amplia mejoría que ofrece estar en un entorno sin redes Wi-Fi que provoquen interferencias, ya que a pesar de que la medida realizada en el entorno ideal los ordenadores receptor y transmisor estén juntos como el caso del entorno ideal, habiendo una distancia mínima, ya la sola presencia de inferencias reduce el Throughput en 18 Mbps (100 Mbps → 82 Mbps) y eso si solamente se tiene en cuenta el mejor caso, incrementándose la diferencia para los demás casos.

El overhead obtenido para la mejor medida obtenida en una red Ad-hoc en un entorno doméstico sabiendo que el Throughput es de 82 Mbps y una velocidad de transmisión es de 0,37.

Para los demás casos, el overhead siempre será mayor. Debido a la reducción a la velocidad de transmisión causada ya sea por el incremento de la distancia entre el receptor y el transmisor o el aumento de las interferencias con otras redes Wi-Fi, además de provocar obviamente una reducción del Throughput (Throughput < Velocidad de transmisión), aumentará como ya se mencionó previamente el número de A-MPDUs, ya que cada A-MPDU contendrá una menor cantidad de tramas MPDU por lo que el número de A-MPDUs se incrementará y con ello el número de Block ACKs y añadiendo además el incremento de la probabilidad de que se produzca una retransmisión.

El mayor Overhead conseguido en este caso corresponde a la medida 4 del escenario Ad-hoc tarde lejos donde la velocidad de transmisión media es 35,536 Mbps y el Throughput obtenido es 10,23 Mbps. El Overhead resultante es 0,7121.

## Capítulo 7. Conclusiones y líneas de trabajo futuro

---

El trabajo verifica que a pesar del empeoramiento de los escenarios como puede ser cuando se alejan el receptor y el transmisor en una red Wi-Fi Ad-hoc o cuando el receptor y transmisor se alejan del router en una red en modo infraestructura, ambos casos por la tarde, la pérdida de paquetes es prácticamente mínima. En el caso de las medidas realizadas en la red Wi-Fi en modo Ad-hoc a pesar de haber una amplia distancia entre ellos siendo casi el límite necesario para que se realice la medida sin ningún problema no se produjo ninguna pérdida de paquete, debido a la gran cantidad de retransmisiones que permite el mecanismo de retransmisión del chipset de Atheros, lo cual se refleja en el Frame Error Rate (FER).

Esto prueba la robustez y eficacia del estándar 802.11n, ya que con una vista global mirando detenidamente los peores se comprueba que el porcentaje de error no es demasiado alto y el throughput siempre tiene un valor adecuado alrededor de los 10 Mbps en el peor caso, siendo el mejor caso 80 Mbps. Con Iperf se calcula el Jitter y el mejor valor se obtiene en una red Ad-hoc y el peor en una red en modo infraestructura, teniendo en cuenta que el router hace de intermediario y en mi caso la distancia de la ruta de transmisión de datos es mayor en este tipo de red Wi-Fi.

Todas estas medidas se han realizado en un entorno doméstico que puede trasladarse a cualquier entorno doméstico con una conexión Wi-Fi 802.11n básica ya que todas las medidas realizadas se han empleado las características básicas del 802.11n: 2 spatial streams, intervalo de guarda largo y un canal Wi-Fi de 10 MHz.

En líneas de trabajo futuro se podría continuar este mismo trabajando en el mismo estándar 802.11n, pero ampliando las características de la red para conseguir mayores velocidades de transmisión, por ejemplo, con canales Wi-Fi de 40 MHz, empleando el intervalo de guarda corto o aumentando el número de spatial streams, siendo la máxima velocidad de transmisión que se puede conseguir ahora con el estándar 802.11n es hasta 600 Mbps o incluso haciendo distintas medidas en la banda de los 2,4 GHz y los 5 GHz.

Otra línea de trabajo futuro posible sería la realización de este mismo trabajo, pero con el estándar 802.11ac que es una mejora del estándar 802.11n que solo trabaja en la banda de los 5 GHz. Este nuevo estándar es compatible con el estándar 802.11n y lleva ya un tiempo en el mercado. Además, incorpora varias mejoras como canales de una anchura de 80 o 160 MHz, una modulación de alta densidad (256 QAM), puede incorporar hasta 8 spatial streams y la tecnología MU-MIMO que permite varias transmisiones simultáneas a los usuarios. Con todo ello se permite una velocidad de transmisión superiores a 1 Gbps.

Con todas estas posibles líneas de trabajo futuro, sumado al hecho de que está previsto para este 2019 la puesta al público del estándar 802.11ax, el cual será una mejora del estándar 802.11ac, sería interesante ver una sucesión de trabajos de estos estándares en el que se realicen medidas, se obtengan resultados y se vea la evolución de las redes Wi-Fi.

## Bibliografía

---

- [1] Matthew S. Gast "802.11n Wireless Networks: a survival guide"; O'Reilly, 2013
- [2] R. Karmakar, S. Chattopadhyay, S. Chakraborty, "Impact of IEEE 802.11n/ac PHY/MAC High Throughput Enhancements on Transport and Application Protocols – A Survey", IEEE Communications Surveys and Tutorials, DOI 10.1109/COMST.2017.2745052, 2017
- [3] R. K. Sheshadri and D. Koutsonikolas, "On packet loss rates in modern 802.11 networks," *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, Atlanta, GA, 2017, pp. 1-9. doi: 10.1109/INFOCOM.2017.8057130
- [4] Lara Deek, Eduardo García-Villegas, Elizabeth Belding, Sung-Ju Lee, Kevin Almeroth, A practical framework for 802.11 MIMO rate adaptation, *Computer Networks*, Vol 83, 2015, pp 332-348, ISSN 1389-1286 <https://doi.org/10.1016/j.comnet.2015.03.015>.
- [5] Z. Shah, S. Rau and A. Baig, "Throughput comparison of IEEE 802.11ac and IEEE 802.11n in an indoor environment with interference," *2015 International Telecommunication Networks and Applications Conference (ITNAC)*, Sydney, NSW, 2015, pp. 196-201. doi: 10.1109/ATNAC.2015.7366812